



(51) International Patent Classification:

G06Q 20/34 (2012.01) G06Q 20/40 (2012.01)  
G06Q 20/32 (2012.01)

(21) International Application Number:

PCT/CA2013/050294

(22) International Filing Date:

16 April 2013 (16.04.2013)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

61/624,947 16 April 2012 (16.04.2012) US  
61/673,096 18 July 2012 (18.07.2012) US  
61/713,302 12 October 2012 (12.10.2012) US

(71) Applicant: SALT TECHNOLOGY INC. [CA/CA]; 56 The Esplanade, Suite 220, Toronto, Ontario M5E 1Z4 (CA).

(72) Inventors: LAW, Simon; 903 Vintner Drive, Mississauga, Ontario L4W 4S6 (CA). SHVARTSMAN, Michael; 6100 Aqua Avenue, Miami, Florida 33141 (US). ROBERGE, Pierre Antoine; 20 Woburn Avenue, Toronto, Ontario M5M 1K6 (CA). DUONG, Peter Thien; 4183 Lingfield Crescent, Mississauga, Ontario L4W 3M3 (CA).

(74) Agents: SO, Wilfred P. et al.; Blake, Cassels & Graydon LLP, Commerce Court West, 199 Bay Street, Suite 4000, Toronto, Ontario M5L 1A9 (CA).

(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, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, 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, 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, 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, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— of inventorship (Rule 4.17(iv))

Published:

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

[Continued on next page]

(54) Title: SYSTEMS AND METHODS FOR FACILITATING A TRANSACTION USING A VIRTUAL CARD ON A MOBILE DEVICE

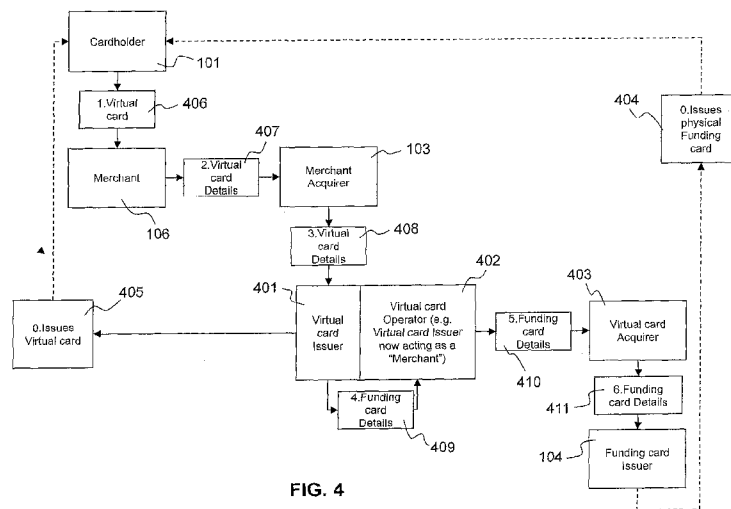


FIG. 4

(57) Abstract: Systems and methods are provided for facilitating contactless payment using cloud- based wallet containing payment credentials (e.g. Visa, Mastercard, American Express) using a near field communication (NFC)-capable device and payment gateway servers. A user can use their existing payment card, herein referred to as a funding card, for contactless payments. A second payment card, herein referred to as a virtual card, is generated. The virtual card is associated with the funding card on a payment gateway server. The virtual card is used on a NFC-enabled mobile device. When a payment is initiated, the virtual card data is sent through the NFC system from a point of sale terminal. This information is sent to the payment gateway server, which retrieves the funding card to make the payment. The funding card, not the virtual card, is used to transfer the money to make payment.

WO 2013/155627 A1

- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

1     **SYSTEMS AND METHODS FOR FACILITATING A TRANSACTION USING A VIRTUAL**  
2                                   **CARD ON A MOBILE DEVICE**  
3  
4

5     CROSS-REFERENCE TO RELATED APPLICATIONS

6     **[0001]**     This application claims priority from: United States Provisional Application No.  
7     61/624,947 filed on April 16, 2012, and titled "SYSTEMS AND METHODS FOR MAKING A  
8     PAYMENT USING A MOBILE DEVICE"; United States Provisional Application No.  
9     61/673,096 filed on July 18, 2012, and titled "SYSTEMS AND METHODS FOR  
10    FACILITATING A TRANSACTION USING A VIRTUAL CARD ON A MOBILE DEVICE"; and  
11    United States Provisional Application No. 61/713,302 filed on October 12, 2012, and titled  
12    "SYSTEMS AND METHODS FOR FACILITATING A PARTY-TO-PARTY VALUE  
13    TRANSFER USING A VIRTUAL CARD ON A MOBILE DEVICE". The entire contents of  
14    these applications are herein incorporated by reference.

15    TECHNICAL FIELD

16    **[0002]**     The following relates generally to facilitating a payment transaction at a merchant  
17    location using a virtual card on a mobile device.

18    DESCRIPTION OF THE RELATED ART

19    **[0003]**     Mobile devices can be used to facilitate a payment transaction, for example, in  
20    exchange for a good or service at a merchant store. A mobile device can be equipped with  
21    a near field communication (NFC) system which can be used to transfer the buyer's payment  
22    credential, such as credit card information, to a point of sale terminal that is also equipped  
23    with a NFC-compatible system to complete the payment transaction.

24    BRIEF DESCRIPTION OF THE DRAWINGS

25    **[0004]**     Embodiments will now be described by way of example only with reference to the  
26    appended drawings wherein:

27    **[0005]**     Fig. 1 is a schematic diagram of an example embodiment of a payment system.

28    **[0006]**     Fig. 2 is a schematic diagram of an example embodiment of one side of a funding  
29    credit card.

30    **[0007]**     Fig. 3 is a schematic diagram of the other side of the funding credit card in Fig. 2.

31    **[0008]**     Fig. 4 is a schematic diagram of an example embodiment of a payment system  
32    showing the flow of data when using a virtual card to facilitate a payment.

- 1 **[0009]** Fig. 5 is another schematic diagram view of the entities involved in an example  
2 embodiment of a payment transaction using a virtual card to facilitate a payment.
- 3 **[0010]** Fig. 6 is a block diagram of an example embodiment of a mobile device.
- 4 **[0011]** Fig. 7 is a flow diagram of an example embodiment of computer executable or  
5 processor implemented instructions for generating and using a virtual card in a payment  
6 transaction according to a user's perspective, the instructions performed by at least a mobile  
7 device, a point of sale terminal and a payment gateway server.
- 8 **[0012]** Fig. 8 is a flow diagram of an example embodiment of computer executable or  
9 processor implemented instructions for facilitating payment transaction using a virtual card,  
10 the instructions showing the interaction between at least a merchant and a payment gateway  
11 server.
- 12 **[0013]** Fig. 9a is a flow diagram of an example embodiment of computer executable or  
13 processor implemented instructions for computing virtual card details by the payment  
14 gateway server, the instructions implemented as part of the transaction process of Fig. 7.
- 15 **[0014]** Fig. 9b is a flow diagram of an example embodiment of computer executable or  
16 processor implemented instructions for authorising virtual card details by the payment  
17 gateway server, the instructions implemented as part of the transaction process of Fig. 8.
- 18 **[0015]** Fig. 10 is a flow diagram of an example embodiment of computer executable or  
19 processor implemented instructions for a registration process between a mobile device and  
20 a payment gateway server.
- 21 **[0016]** Fig. 11 is a flow diagram of an example embodiment of computer executable or  
22 processor implemented instructions for computing virtual card details including a primary  
23 account number and discretionary data, using the data exchanged during the registration  
24 process of Fig. 10.
- 25 **[0017]** Fig. 12 is a flow diagram of an example embodiment of computer executable or  
26 processor implemented instructions for verifying the discretionary data of Fig. 11.
- 27 **[0018]** Fig. 13 is a schematic diagram showing example components of a system used  
28 to facilitate an e-commerce transaction using a virtual card.
- 29 **[0019]** Fig. 14 is a screenshot of an example embodiment of a graphical user interface  
30 (GUI) for performing an e-commerce transaction using a virtual card.

1 [0020] Fig. 15 is a screenshot of another example embodiment of a GUI for performing  
2 an e-commerce transaction using a virtual card.

3 [0021] Fig. 16 is a flow diagram of an example embodiment of computer executable or  
4 processor implemented instructions for performing an e-commerce transaction using a  
5 virtual card, the instructions performed by at least a mobile device, a point of sale terminal  
6 and a payment gateway server.

7 [0022] Fig. 17 is a flow diagram of an example embodiment of computer executable or  
8 processor implemented instructions for performing an e-commerce transaction using a  
9 virtual card, the instructions showing the interaction between at least a merchant and a  
10 payment gateway server.

11 [0023] Fig. 18 is a flow diagram showing a party-to-party value transfer.

12 [0024] Fig. 19 is a schematic diagram showing example components of a system used  
13 to facilitate a party-to-party value transfer using a transfer ID.

14 [0025] Fig. 20 is an example embodiment illustrating a sender's mobile device and a  
15 receiver's mobile device, the sender's mobile device showing a graphical user interface  
16 (GUI) for selecting a funding card.

17 [0026] Fig. 21 is an example embodiment illustrating the sender's mobile device and the  
18 receiver's mobile device of Fig. 20 prior to being "tapped" together to facilitate a party-to-  
19 party value transfer.

20 [0027] Fig. 22 is an example embodiment illustrating the sender's mobile device and the  
21 receiver's mobile device, after the tapping in Fig. 21, with both devices showing a GUI  
22 indicating that the party-to-party value transfer has been completed.

23 [0028] Fig. 23 is a flow diagram of an example embodiment of computer executable or  
24 processor implemented instructions for computing a transfer ID by the payment gateway  
25 server and the mobile device, such that the transfer ID is used facilitate the transfer of value.

26 [0029] Fig. 24 is a flow diagram of an example embodiment of computer executable or  
27 processor implemented instructions for validating the transfer ID and issuing a prepaid virtual  
28 card to the receiver, thereby completing the party-to-party value transfer, the instructions  
29 implemented as a continuation of process of Fig. 23.

30 [0030] Fig. 25 is a schematic diagram showing example components of a system used  
31 to facilitate a party-to-party value transfer using a virtual card.

1 [0031] Fig. 26 is a flow diagram of an example embodiment of computer executable or  
2 processor implemented instructions for computing virtual card details by the payment  
3 gateway server and the mobile device.

4 [0032] Fig. 27 is a flow diagram of an example embodiment of computer executable or  
5 processor implemented instructions for authorising virtual card details and completing the  
6 party-to-party value transfer, the instructions implemented as a continuation of process of  
7 Fig. 26.

8 [0033] Fig. 28 is a flow diagram of an example embodiment of computer executable or  
9 processor implemented instructions for a registration process between the sender's mobile  
10 device and a payment gateway server.

11 [0034] Fig. 29 is a flow diagram of an example embodiment of computer executable or  
12 processor implemented instructions for computing virtual card details including a primary  
13 account number and discretionary data, using the data exchanged during the registration  
14 process of Fig. 28.

15 [0035] Fig. 30 is a flow diagram of an example embodiment of computer executable or  
16 processor implemented instructions for verifying the discretionary data of Fig. 29.

#### 17 DETAILED DESCRIPTION

18 [0036] It will be appreciated that for simplicity and clarity of illustration, where considered  
19 appropriate, reference numerals may be repeated among the figures to indicate  
20 corresponding or analogous elements. In addition, numerous specific details are set forth in  
21 order to provide a thorough understanding of the example embodiments described herein.  
22 However, it will be understood by those of ordinary skill in the art that the example  
23 embodiments described herein may be practiced without these specific details. In other  
24 instances, well-known methods, procedures and components have not been described in  
25 detail so as not to obscure the example embodiments described herein. Also, the  
26 description is not to be considered as limiting the scope of the example embodiments  
27 described herein.

28 [0037] Fig. 1 shows an example embodiment of a typical in-person payment flow  
29 between a cardholder 101, a merchant 102, a merchant acquirer 103, a payment network  
30 109 and a card issuer 104. The merchant 102, the merchant acquirer 103, the payment  
31 network 109 and the card issuer are each associated with computing devices including  
32 processors and memory. By way of background, this payment flow is also known as a  
33 standard 4-party model, or a standard 4-party protocol. The card issuer 104 is typically a

1 bank which is associated with a payment network. It facilitates the cardholder's use of the  
2 payment card to pay for goods or services based on the cardholder's promise to repay the  
3 card issuer for the purchase. Non-limiting examples of payment networks include Visa,  
4 MasterCard, American Express and Diners Club/Discover. The card issuer 104 issues the  
5 payment card, which is associated with a payment network 109, to the cardholder 101 (block  
6 105). It is understood that all references to credit card would also apply to debit card,  
7 prepaid card, or other payment credentials used to facilitate payment at a merchant location.  
8 The term "funding card" is used herein to refer to debit cards, credit cards, prepaid cards,  
9 and the like.

10 **[0038]** To make a purchase or payment, the cardholder 101 (e.g. a person) provides  
11 funding card information to the merchant 102 (block 106). For example, the funding card  
12 can be "swiped" using a magnetic stripe card reader device, or the funding card number can  
13 be read to the merchant 102. The merchant 102 (e.g. a computing device) sends the  
14 payment authorisation, including the funding card details, to a merchant acquirer 103 (block  
15 107). An acquirer 103 is an organization that collects payment-authorisation requests from  
16 merchants and facilitates the payment transaction with the payment network on behalf of the  
17 merchants.

18 **[0039]** When the acquirer 103 gets the funding card payment authorisation request, the  
19 acquirer submits the payment transaction (block 108) to the payment network 109.

20 **[0040]** When the payment network 109 receives the funding card payment authorisation  
21 request, the payment network forwards the payment transaction (block 110) to the card  
22 issuer 104 for payment authorisation.

23 **[0041]** When the card issuer 104 receives the payment transaction, it checks the  
24 transaction details for validity. This includes checking the card number, the expiration date,  
25 the funding card limit, etc. The card issuer 104 responds to the merchant acquirer 103 with a  
26 payment authorisation code (approved or declined response, transaction identification  
27 number, etc) via the payment network 109. The merchant acquirer 103 forwards the  
28 response to the merchant 102. Then the merchant 102 shares the payment authorisation  
29 results with the cardholder.

30 **[0042]** It is recognized that such an established computer payment system is well-  
31 established and is adopted by many users and large companies. It is recognized however,  
32 that such a computer payment system does not provide data processing means for making  
33 payments on a mobile device. Such a computer payment system also requires that the

1 funding card details pass through the merchant 102 and the merchant acquirer 103, which  
2 can expose or reveal sensitive financial data. This security risk posed by such a computer  
3 payment system is undesirable.

4 **[0043]** By way of background, an example embodiment of a credit card 201 is shown in  
5 Fig. 2 and Fig. 3. One side of the credit card, as per Fig. 2, shows a logo 202 or mark  
6 identifying the card issuer. Typically, the card issuer is a bank. A logo 203 or mark is also  
7 shown to identify the payment network. The card 201 also shows the credit card number  
8 204. The credit card number 204 is sometimes referred to as a primary account number  
9 (PAN). The length and format of the credit card number 204 varies depending on the card  
10 issuer and the payment network. Generally, the first digit in the credit card number 204  
11 identified the payment network. The last digit of the credit card number 204 is a check digit.  
12 A check digit is a form of redundancy check used for error detection. The intermediary  
13 numbers can signify a bank number (e.g. associated with the card issuer) and an account  
14 number (e.g. of the bank).

15 **[0044]** In the example shown in Fig. 2, as per block 207, the first digit "4" identifies that  
16 Visa is the payment network. Digits two through six are the issuing bank identification  
17 number. Digits seven through fifteen are the account number. Digit sixteen is the check  
18 digit.

19 **[0045]** The credit card 201 also includes a range of dates 205 that the credit card is  
20 valid. The range of dates includes an expiry date of the credit card. The name 206 of the  
21 cardholder is also shown.

22 **[0046]** In Fig. 3, on the other side of the credit card 201, there is a magnetic strip 301,  
23 the signature 302 of the cardholder's name, and a static security code 303. The static  
24 security code is printed on the credit card 201.

25 **[0047]** The magnetic stripe 301, also sometimes referred to as a magstripe, is typically  
26 made up of tiny iron-based magnetic particles in a plastic-like film. The stripe 301 has  
27 information that is written on it.

28 **[0048]** Often there are two or three tracks on data encoded on the stripe 301. In an  
29 example embodiment, each track is about one-tenth of an inch wide. The ISO/IEC standard  
30 7811, which is used by banks, specifies that track one is 210 bits per inch (bpi), and holds 79  
31 6-bit plus parity bit read-only characters; track two is 75 bpi, and holds 40 4-bit plus parity bit  
32 characters; and track three is 210 bpi, and holds 107 4-bit plus parity bit characters. Other  
33 data formats may apply.



1 **[0049]** In an example embodiment, the format for Track Two, developed by the banking  
2 industry, is as follows (maximum of 40 characters):

- 3 • Start sentinel - one character (generally ';')
- 4 • Primary account number - up to 19 digits
- 5 • Separator - one character (generally '=')
- 6 • Expiration date - four digits in the form of the last two digits of the year, and the two digits  
7 representing the month (e.g. YYMM)
- 8 • Service Code – three digits. The first digit specified the interchange rules, the second  
9 specifies authorisation processing and the third specifies the range of service
- 10 • Discretionary data – balance of available space to fill the length of the Track Two (ex 10  
11 digits if the PAN is 19 digits long)
- 12 • End sentinel – one character (generally '?')
- 13 • Longitudinal Redundancy Check (LRC) - one character

14 **[0050]** The above data is herein generally referred to as Track Two data.

15 **[0051]** It is recognized, that in addition to the typical plastic card form factor, as shown in  
16 Fig.2 and Fig. 3, there are a growing numbers of other form factors that can be used for  
17 payment. Non-limiting examples include a mini card, fob, mobile device, etc. It is also  
18 appreciated that other types of funding cards, in addition to credit cards, include similar  
19 information (e.g. PAN, expiry date, Track Two data, etc.).

20 **[0052]** The way the card data itself is stored on a funding card and shared with a  
21 payment terminal has many variants, such as storing the card data on a magnetic stripe and  
22 reading the data using a magnetic card reader, or storing the card data using chip  
23 technology and using a chip-reader compatible reader to interact with the card. For  
24 example, although not shown in Fig. 2 and Fig. 3, the credit card can include an integrated  
25 circuit or logic chip capable of performing computations. Various examples of chip-based  
26 card technology includes contactless magnetic stripe emulation, EMV contact, EMV  
27 contactless, etc. EMV stands for Europay, MasterCard and VISA, a global standard for  
28 inter-operation of integrated circuit cards (IC cards or "chip cards") and IC card reader. In  
29 addition to facilitate the exchange of information via the contact interface of the card,  
30 information can also be exchanged using the contactless capabilities of capable card. Such  
31 contactless cards are sometimes referred to as PayPass, payWave, and ExpressPay.

32 **[0053]** It is recognized that contactless payment is well suited for a mobile device. For  
33 example, some mobile devices have a near-field communication (NFC) system that allows  
34 data to be transferred wirelessly, using the ISO/IEC 18092 standard or other compatible  
35 standards, over a relatively short distance. A NFC-enabled mobile device can establish

1 radio communication with another contactless-enabled device by touching them together or  
2 keeping them into close proximity. The term “mobile device” and “mobile phone” are  
3 interchangeably used herein.

4 **[0054]** An NFC-enabled mobile device can be equipped with a “software payment card”  
5 (e.g. a software implementation of a contactless payment card) application running inside  
6 the mobile device. The software payment card uses the compatible contactless  
7 communication capabilities of the device to interact with contactless-enabled POS terminal  
8 to facilitate payment transactions.

9 **[0055]** In many cases, the software payment card is stored on the secure element to  
10 protect the sensitive payment data such as the one used to generate the dynamic data  
11 necessary to complete a typical contactless payment transaction. It is recognized that  
12 secure elements are limited in their storage capabilities. They are typically available in the  
13 form of an embedded chip, such as in a universal integrated circuit card (UICC), or in a  
14 subscriber identity module (SIM) card.

15 **[0056]** In part to maintain the security of the various applications running inside the  
16 secure element, the secure element is typically managed by a mobile phone carrier  
17 distributing the secure element with the mobile device. Part of the managed service includes  
18 delivering applications into the secure element directly, or giving permission to a third party  
19 organization to deploy their application on a particular secure element. The managed  
20 service is typically delivered using what the industry referred to as a trusted service manager  
21 (TSM).

22 **[0057]** All applications stored and running inside the secure element, such as the  
23 individual “software payment card”, need their own space. Payment cards are issued to  
24 consumers by the card issuer. Deploying software payment cards on mobile phones requires  
25 a high level of coordination between the mobile phone carrier and the card issuer where the  
26 mobile phone carrier provides access to individual secure elements, one at a time, to the  
27 issuer. Only cards from funding card issuers that have the infrastructure and the agreement  
28 with the mobile phone carrier can be used on the mobile phone for contactless payment.  
29 This is limiting for both the card issuers and for cardholders.

30 **[0058]** In the United States, for example, there are thousands of issuing banks and tens  
31 of mobile phone carriers. The mobile phone carriers use the TSM, in part, to issue and  
32 manage payment credentials into mobile phone. The TSM enables mobile phone carriers  
33 (also called service providers) or other entities controlling the secure element on mobile

1 phones to distribute and manage remotely the secure applications running inside the secure  
2 element by securing access to the secure element in mobile-enabled devices. An example  
3 of secure application is a software payment card. The issuing banks also interact with the  
4 mobile phone carriers through a service (in an example embodiment, called ISIS) to manage  
5 payment credentials issued to NFC-enabled devices. As a non-limiting example, in Canada,  
6 Rogers Communication is a mobile carrier that is in partnership with the Canadian Imperial  
7 Bank of Commerce (CIBC), an issuing bank, to issue software payment card to CIBC  
8 cardholder's phone on the Rogers network. Establishing systems and methods to manage  
9 the secure issuance and management of software payment cards between the issuing  
10 banks, the mobile phone carriers, and the mobile device is costly and complex. It involves  
11 pre-arranged agreements between the parties as well as some customizations of the  
12 software and computing systems to meet the needs of all the entities participating. This can  
13 make it difficult for smaller-sized card issuers to adopt NFC technology for facilitate  
14 payments. It also means that the issuing bank will likely require numerous point-to-point  
15 connections with the various mobile operators it wants to supports, which requires further  
16 hardware capabilities and software to support and organize data communications between  
17 different mobile operators. The systems and methods described herein attempt to address  
18 at least one of these issues.

19 **[0059]** It is also recognized that from the user's perspective, the process of associating  
20 their mobile device with their funding card to be used for contactless payment is very much  
21 dependent on pre-arranged relationships between the mobile phone carrier and the card  
22 issuers. Therefore, a user has limited options or no options when determining if their current  
23 funding card can be associated with their mobile phone for contactless payments. For  
24 example, a user has a funding card from Funding Card Issuer A. The user also has a NFC-  
25 enabled mobile phone from Mobile Phone Carrier B. However, Mobile Phone Carrier B only  
26 has a pre-arranged agreement and infrastructure to facilitate contactless payments with  
27 Funding Card Issuer B. Therefore, even if the user wanted to use their mobile phone to  
28 make a contactless payment, the user would not be able to because there is no pre-  
29 arranged agreement and computer network infrastructure between Mobile Phone Carrier B  
30 and Funding Card Issuer A to issue a software payment card into the user's phone. This  
31 limitation of the computer network infrastructure limits the user's ability to make NFC-type  
32 payments with their mobile device.

33 **[0060]** In a typical example embodiment of making a payment using an NFC-enabled  
34 mobile device, a user first requests to its issuer to load a funding card on their mobile device.

1 When this process is completed, a software funding card has been delivered and installed  
2 securely on the secure element on the user's mobile device. The user can now use the  
3 funding card on the mobile device's to facilitate contactless payment transactions at  
4 merchant locations that are equipped to accept such type of payment transaction. Once the  
5 user attempts to pay with their mobile device at a merchant, the payment details are sent  
6 and are verified by the funding card issuing server in a similar manner to other payment  
7 transaction.

8 **[0061]** When a contactless funding card credential is used to facilitate a payment  
9 transaction, the transaction most often includes dynamic data from the card to securely  
10 authenticate the payment credential. The dynamic data changes values every time the  
11 credential is used. If the dynamic data received on the funding card issuing server matches  
12 the expected value computed by the server for the card, the authentication of the payment  
13 credentials is deemed successful and the payment authorisation can continue. It can be  
14 appreciated that there are various ways in which the mobile device and the funding card  
15 issuing server can compute the dynamic data used to authenticate the payment credential.

16 **[0062]** In an example embodiment for credit cards, the dynamic data is a rotating card  
17 verification value (CVV, also referred sometime to dynamic CVV or dCVV). This rotating  
18 CVV is computable based on changing information provided by the integrated circuit inside  
19 the card. In another example embodiment, the dynamic data is dynamic EMV data which is  
20 computed using random data from the funding card, or random data from a merchant's point  
21 of sale terminal, or both. A common implementation of dynamic data uses an Application  
22 Transaction Counter (ATC) on the card so that every transaction produces a different data  
23 stream. This is achieved as the ATC is incremented by '1' for every transaction performed.  
24 When a user taps, touches or positions their mobile device near a contactless-enabled point  
25 of sale (POS) terminal, the funding card data (card number, dynamic data, expiry date, etc.),  
26 hereafter referred to as the card's Track Two data, is sent from the mobile device to the POS  
27 terminal. This information then makes it way to the funding card issuing server for  
28 verification. The funding card issuer will perform numerous checks to validate the  
29 transaction, including comparing the dynamic data of the mobile device with the value  
30 generated by the server. If the dynamic data is matching, and all the other checks and  
31 controls performed by the funding card issuer are successful, the funding card issuer will  
32 respond with a positive payment authorisation response.

33 **[0063]** It is also recognized that a card application specific to a given funding card can  
34 be installed on the mobile device and used to interact with POS terminal as described

1 above. It is also recognized that the card application is typically installed on the mobile  
2 device's secure element. Typically, each funding card has its own corresponding card  
3 application that resides on the mobile device's secure element. It can be appreciated that as  
4 each card application takes up storage space on the secure element, and that the secure  
5 element typically has very limited storage space, having multiple card applications on the  
6 secure element in some cases is not possible due to insufficient storage space. By way of  
7 background, the secure element can have a native operating system that be programmed  
8 to perform various tasks and activities, including for example, a card application that  
9 emulates the magnetic strip data of a funding card or a card application that emulates the  
10 data used in an EMV contactless payment. Also by way of background, and by way of  
11 example, a typical secure element has memory of 256 kB, and each card application can  
12 consume memory of 40-80 kB. It can therefore be appreciated that associating multiple  
13 funding cards (and each of their card applications) with a mobile device for NFC payments  
14 can be limiting.

15 **[0064]** Therefore, it is desirable to reduce the amount of storage space that card  
16 applications require on the secure element so as to not limit the number of software payment  
17 cards a user can load into a secure element. Along the same lines, it is desirable for mobile  
18 phone carriers to reduce the amount of data used by "card application" on the secure  
19 element so that other types of applications can be loaded thereon. It is also desirable to  
20 reduce costs incurred by the funding card issuer to issue and operate software payment  
21 cards into secure elements. By way of background, a mobile phone carrier typically charges  
22 application providers, such as funding card issuers, for the amount of storage space used on  
23 the secure element. It is also desirable to reduce the amount of infrastructure required by  
24 the funding card issuer to issue a software payment card for the mobile phone. It is also  
25 desirable to reduce the amount of coordination required between the funding card issuer and  
26 the mobile phone carrier to issue a software payment card on particular mobile phone. It is  
27 also desirable to enable the user (e.g. the cardholder) to load any, and as many, funding  
28 cards they want into their NFC-enabled mobile phone, independently of the funding card  
29 issuer having the infrastructure or a commercial relationship or agreement with a particular  
30 mobile phone carrier.

31 **[0065]** The systems and methods described herein attempt to address the above issues.

32 **[0066]** It will be appreciated that different features of the example embodiments of the  
33 proposed systems and methods, as described in this document, may be combined with each  
34 other in different ways. In other words, a feature described with respect to one embodiment

1 of a mobile payment system or method can be applied to another embodiment of the mobile  
2 payment system or method, although not specifically stated.

3 **[0067]** In general, the systems and methods described herein allow a cloud-based wallet  
4 payment gateway server to synchronize with a NFC-enabled mobile device and application  
5 to facilitate contactless payment transactions at a merchant store accepting contactless  
6 payment. A pre-arranged agreement or additional infrastructure between the funding card  
7 issuer and the mobile phone carrier is not required. A user selects a funding card for making  
8 the contactless payment, through their mobile device. A second card, herein referred to as a  
9 virtual card, is generated along with all required card data (e.g. PAN, expiry date, dynamic  
10 data, discretionary data, etc.) to complete the payment transaction. The virtual card is  
11 associated with the funding card on the payment gateway server. In an example  
12 embodiment, the association between the virtual card and funding card is limited to some  
13 period of time.

14 **[0068]** In an example embodiment, the data required to compute virtual card data set is  
15 sent to the mobile device, typically every time that the user wants to make a payment. For  
16 example, a new virtual card is created for each and every payment transaction. In another  
17 example, a new virtual card is created based on time limits, or based on certain events, or  
18 both, and has a much shorter usage period compared to a standard funding card which can  
19 be used typically over several years. Where the virtual card can be used with the funding  
20 card for several transactions, a data for computing a new virtual card does not need to be  
21 sent to the mobile device each time the user makes a payment.

22 **[0069]** When a payment is initiated, the virtual card data is sent through the NFC system  
23 on the mobile device to a contactless-enabled POS terminal. This information is sent from  
24 the POS terminal to the merchant system; from the merchant system to the merchant  
25 acquiring bank; and from the acquiring bank to the cloud-based wallet payment gateway  
26 server (also referred to the "payment gateway server") via the payment network. The cloud-  
27 based wallet payment gateway server acts as the virtual card issuer server and verifies the  
28 virtual card. If successfully verified, the funding card details associated with the virtual card  
29 are retrieved and sent to the funding card issuer server via the payment network to complete  
30 the payment authorisation transaction. The funding card issuer server verifies the funding  
31 card and sends back an authorization code to the payment gateway server. The payment  
32 gateway server sends back a corresponding authorization code to the merchant system.

33 **[0070]** The payment can be settled when the merchant system initiates a settlement  
34 request, typically, though not necessarily, at the end of every business day. To complete the

1 settlement, the merchant system sends all the virtual card numbers and the corresponding  
2 authorization codes received during the period to the virtual card issuer, via the merchant  
3 acquiring bank. The virtual card issuer verifies the virtual card numbers and authorisation  
4 codes. For all the matching records, the virtual card issuer retrieves the associated funding  
5 card numbers and authorisation codes and sends the data to the funding card issuer for  
6 settlement via the virtual card issuer acquiring bank. The funding card issuer verifies the  
7 funding card numbers and authorization codes, and if successfully verified, sends the money  
8 via a standard method to the payment transaction originator, in this case the virtual card  
9 issuer. At that point, the virtual card issuer sends the money to the merchant acquiring bank  
10 also using a standard method. In an example embodiment, the virtual card issuer is the  
11 payment gateway server, or a module within the payment gateway server.

12 **[0071]** In an example embodiment, the systems and methods described herein allow a  
13 cloud-based wallet (e.g. associated with a funding card) to be synchronized with a NFC-  
14 enabled mobile device (e.g. associated with a virtual card). In another example  
15 embodiment, the systems and methods described herein provide real time rotation of a  
16 primary account number (PAN) and dynamic data used to facilitate contactless payment  
17 transactions. In another example embodiment, the systems and methods described herein  
18 use virtual payment credentials to complete a brick-and-mortar contactless purchase  
19 transaction without disclosing the user's funding card details.

20 **[0072]** Turning to Fig. 4, an example embodiment showing the flow of payment data  
21 using a virtual card is provided. A card issuer, also called the funding card issuer 104,  
22 issues a standard card account, most often in the form of a plastic card, to the user 101  
23 (block 404). The standard card is the funding card, and the user becomes the cardholder.

24 **[0073]** Although not shown, the cardholder 101 registers one or many funding cards and  
25 their mobile device with a virtual card issuer 401. It can be appreciated that any funding  
26 card can be registered, and is not limited or dependent on the mobile phone carrier having  
27 an agreement with the funding card issuer. In other words, even if the funding card and the  
28 mobile phone carrier do not have any agreement or connecting computer infrastructure,  
29 according to the proposed systems and methods, the user's one or many funding cards and  
30 the user's mobile device can be registered with the virtual card issuer 401. It can also be  
31 appreciated that any number of funding cards can be registered in association with the  
32 mobile device. The cardholder's mobile device includes a payment application that can  
33 interact with the virtual card issuer 401.

1 **[0074]** In an example embodiment of the registration, for each funding card the user  
2 wishes to register, the user enters in (e.g. types in) card details into the mobile device (e.g.  
3 card details include the name printed on the funding card, the PAN printed on the funding  
4 card, the expiry date printed on the funding card, and the static security code printed on the  
5 funding card). The mobile device sends this data, plus a user provided PIN and the mobile  
6 device ID to the payment gateway server. For each funding card, the payment gateway  
7 server computes a funding card identifier which identifies the given funding card. The  
8 payment gateway server stores the funding card identifier in association with the funding  
9 card details, mobile device ID and PIN, and it sends the funding card identifier to the mobile  
10 device for storage. In an example embodiment, the funding card identifier is a value that is  
11 different from the PAN, expiry date or static security code of the funding card. For example,  
12 the funding card identifier is a random value so that, if intercepted by an adversary, would  
13 not be able to recognize any funding card details. In an example embodiment, the mobile  
14 device does not store any funding card details but only stores limited funding card details  
15 (e.g. the name funding card issuer and the last 4 digits of the PAN). The mobile device  
16 stores the funding card identifier, which it sends to the payment gateway server to indicate a  
17 specific funding card. It can be appreciated that there are other methods to capture the  
18 funding card details (e.g. besides the user typing in the data), which can be used with the  
19 principles described herein.

20 **[0075]** It can be appreciated that a single payment application is required on the mobile  
21 device, which can manage multiple funding cards. If multiple funding cards are registered,  
22 each of the associated funding card identifiers are stored on the mobile device, within the  
23 single payment application. The details of each individual funding card are stored on the  
24 payment gateway server. In this way, the payment gateway server acts as a cloud-based  
25 server that stores the details of multiple funding cards. Additional details of a funding card  
26 registration process are described below with respect to Fig. 10.

27 **[0076]** Continuing with Fig. 4, when the registered cardholder 101 wishes to make a  
28 contactless payment using their mobile device, the user selects the funding card. The  
29 selected funding card is conveyed to the virtual card issuer 401, and the virtual card issuer  
30 401 issues a virtual card to the cardholder 101, and more specifically to the payment  
31 application running on the mobile device (block 405). The cardholder touches their mobile  
32 device on the merchant system and the virtual card details are delivered to the merchant  
33 106, via a NFC-enabled POS terminal (block 406). The merchant sends the virtual card  
34 details to the merchant acquirer 103 (block 407). Based on some of the virtual card details,



1 the merchant acquirer 103 sends the virtual card details to the virtual card issuer 401 (block  
2 408) via the payment network. The virtual card issuer 401 uses the virtual card details to  
3 determine the corresponding funding card (block 409). The virtual card operator 402 (e.g.  
4 the virtual card issuer now acting as a “merchant”) sends the corresponding funding card  
5 details, including the original transaction amount, to its acquirer 403 (block 410). This  
6 parallels the typical process of a merchant 106 sending the funding card details to the  
7 merchant acquirer 103. The virtual card acquirer 403 sends the funding card details to the  
8 funding card issuer 104 (block 411) via the payment network for a standard payment  
9 authorisation.

10 **[0077]** The funding card issuer 104 then can authorise, or not, the payment request and  
11 responds back with an authorisation code to the virtual card issuer’s acquirer, via the  
12 payment network, which in turn notify the virtual card issuer. The virtual card issuer notifies  
13 the merchant acquirer using a corresponding authorization code for the merchant system via  
14 the payment network.

15 **[0078]** To settle the funds (not shown), the merchant will initiate, or the merchant  
16 acquirer will initiate on behalf of the merchant, a settlement request to the payment network  
17 via the merchant acquirer. The payment network will forward the corresponding virtual card  
18 transactions to settle to the virtual card issuer. When the virtual card issuer receives the  
19 transactions, the virtual card issuer will send a settlement request for the corresponding  
20 funding card transactions to its own acquirer. The settlement request will be sent to the  
21 appropriate funding card issuer by the payment network. The funding card issuer receives  
22 the transactions settlement request, and will send the funds associated with all matching  
23 transactions to the payment transaction originator, in this case the virtual card issuer. When  
24 the virtual card issuer receives the funds, the virtual card issuer will send the corresponding  
25 funds to the merchant.

26 **[0079]** In an example embodiment, the virtual card issuer 401, the virtual card operator  
27 402, and the virtual card acquirer 403 are represented by the same entity, referred herein as  
28 the payment gateway server. The payment gateway server itself can include one or more  
29 servers. For example, each of the entities 401, 402, 403 can be individual servers that,  
30 combined, form the payment gateway server.

31 **[0080]** Turning to Fig. 5, example embodiment components of a system for facilitating  
32 payment using the virtual card is shown. The user 101 has one or more funding cards 505  
33 pre-registered into its cloud-wallet (process not shown). For example, the user has multiple  
34 funding cards. The user 101 also owns an NFC-enabled mobile device 501, which includes

1 a payment application. The mobile device 501 is configured to interact, via NFC, with a  
2 merchant POS terminal device 502. The merchant POS terminal device 502 is in data  
3 communication via the merchant payment system with the merchant acquirer 103 (e.g. a  
4 server). The merchant acquirer and the payment gateway server 506 are in communication  
5 with a payment network (e.g. Visa, MasterCard, Discover, etc.), which is configured to send  
6 data related to payments and transactions to relevant parties, including the payment  
7 gateway server 506 and the funding card issuer 104 (e.g. a server). The communication  
8 between the merchant POS terminal device 502, the merchant acquirer 103, the payment  
9 network 504, the payment gateway server 505 and the funding card issuer 104 can occur  
10 over wired or wireless communication networks, or both.

11 **[0081]** The payment gateway server 506 is also in communication with the mobile device  
12 501 through a wireless network. For example, the wireless network is provided by a mobile  
13 phone carrier.

14 **[0082]** Example components of the payment gateway server 506 are shown in block  
15 507. The server 506 can include the virtual card issuer 401, the virtual card operator 402  
16 and the virtual card acquirer 403. During a registration process conducted by the user 101,  
17 the payment gateway server 506 stores the user selected personal identification number  
18 (PIN), funding card data in association with the user's mobile device ID (e.g. in database  
19 508). For example, the user PIN, funding card1 and funding card2 (and other funding cards)  
20 are stored in association with the mobile device ID of mobile device 501. Also, in database  
21 509, the temporary data associations between a given funding card, a virtual card and  
22 authorisation code when applicable are stored. For example, virtual card1, funding card1  
23 and authorisation code 1 are all stored in association with each other's. Another example is  
24 virtual card2 and funding card2 are also associated but have no authorisation code issued  
25 yet for the card pair. In an example embodiment, there is no authorisation code issued yet  
26 because the user has not yet tapped the mobile device using funding card2 (e.g. really  
27 virtual card2), or because the merchant payment authorisation was interrupted such that the  
28 payment gateway server never received the authorisation code.

29 **[0083]** Turning to Fig. 6, example components of the mobile device 501 are shown.  
30 The mobile device 501 includes a main processor 601 which interacts with a number of  
31 components including, among other things, auxiliary inputs/outputs 302, a data port 603, a  
32 keyboard 604, a speaker 605 (e.g. an audio speaker), a microphone 606, a GPS receiver  
33 607 and a camera 608. The mobile device 501 also includes an NFC subsystem 609, a

1 secure element 622 which may or may not also have direct connectivity to the NFC  
2 subsystems 609, and other device subsystems 611.

3 **[0084]** The mobile device 501 uses a communication system 613 to interact with a  
4 wireless network 612. Memory types include flash memory 614 and random access memory  
5 (RAM) 615. The mobile device's display 616 can be a touch-screen type display or another  
6 type of display.

7 **[0085]** An operating system 617 may be used to manage and run software components  
8 618. Software components or applications include a web browser or internet browser 619  
9 and payment application 620. Other software components 621 are included.

10 **[0086]** The secure element 622 can be used for storing information such as application  
11 and data elements, for example a payment application and a mobile identifier. In an example  
12 embodiment, the secure element is inside a subscriber identity module (SIM) card. Non-  
13 limiting examples of mobile devices include cell phones, smart phones, PDAs, tablets,  
14 netbooks, and laptops.

15 **[0087]** It will be appreciated that any module or component exemplified herein that  
16 executes instructions or operations may include or otherwise have access to computer  
17 readable media such as storage media, computer storage media, or data storage devices  
18 (removable and/or non-removable) such as, for example, magnetic disks, optical disks, or  
19 tape. Computer storage media may include volatile and non-volatile, removable and non-  
20 removable media implemented in any method or technology for storage of information, such  
21 as computer or processor readable instructions, data structures, program modules, or other  
22 data, except transitory propagating signals per se. Examples of computer storage media  
23 include RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital  
24 versatile disks (DVD) or other optical storage, magnetic cassettes, magnetic tape, magnetic  
25 disk storage or other magnetic storage devices, or any other medium which can be used to  
26 store the desired information and which can be accessed by an application, module, or both.  
27 Any such computer storage media may be part of the any one of the servers 103, 504, 104,  
28 506, 401, 402, 403, the mobile device 501, the POS terminal 502, etc. or accessible or  
29 connectable thereto. Any application or module herein described may be implemented using  
30 computer or processor readable/executable instructions or operations that may be stored or  
31 otherwise held by such computer readable media.

1 **[0088]** Turning to Fig. 7, example computer executable or processor implemented  
2 instructions are provided for facilitating a transaction between a mobile device 501 and a  
3 POS terminal device 502.

4 **[0089]** It is assumed that the end-user has a compatible NFC-mobile device 501. It is  
5 assumed that payment application 620 has been installed on the mobile device 501. It is  
6 assumed that the end user has registered for the service.

7 **[0090]** Following the merchant capturing the purchase transaction details into its point of  
8 sale system, the NFC-enabled POS terminal device 502 displays a message to the end user  
9 to “tap” to pay (block 701). For example, the user can tap their mobile device 501 or a  
10 contactless funding card. The examples described herein relate to tapping the mobile  
11 device 501.

12 **[0091]** The end user 101 sees the message from the POS terminal device 502 and  
13 starts the payment application 620 on the mobile device 501 (block 702). For example, the  
14 user selects an icon for the payment application 620 on a home screen of the mobile device  
15 501, thereby launching the payment application 620. The payment application 620  
16 determines if the user has successfully registered to the service (block 703), and if so,  
17 shows a menu (block 704) of supported actions by the application. If the user has not  
18 registered, the menu offers the user to register with the service (e.g. register one or more  
19 funding cards, provide a PIN, link a mobile device identifier to the registration record).

20 **[0092]** A graphical user interface (GUI) on the menu is able to receive an input from the  
21 user to initiate a payment with a virtual card with the payment application (block 705).  
22 Example of other menu item includes “add a funding card”, “delete a funding card”, etc.

23 **[0093]** The mobile device 501 then displays the funding cards that have been pre-  
24 registered by the user (block 706). A user input is provided to select one of the funding  
25 cards (block 707). In an example embodiment, the displayed funding card information is  
26 loaded after the end user successfully registers to the service and has registered at least  
27 one funding card. The list is updated when the user adds an additional funding card into the  
28 payment application or when a funding card is deleted. For each registered funding card,  
29 there is a corresponding record stored on the payment application 620 and on the payment  
30 gateway server database 508 which includes an identifier for the payment network  
31 associated with the funding card, a funding card identifier, etc.

32 **[0094]** The payment application 620 sends the mobile device identifier to the payment  
33 gateway server 506 (block 708). The payment application 620 also sends the transaction

1 type (selected action in the application menu, in this case, "Pay with a virtual card") and the  
2 identifier for the selected funding card to the payment gateway server 506 (block 709).  
3 Based on the information received by mobile device 501, the payment gateway server 506  
4 creates and computes the details regarding the virtual card (block 710). In particular, the  
5 payment gateway server 506 computes a virtual PAN, an expiry date of the virtual card and  
6 all or some of the data elements that form the Track Two data. It is noted that Track Two  
7 data includes, among other things: the PAN, a service code, an expiry date, discretionary  
8 data and a LRC. In an example embodiment, the payment gateway server 506 at this time  
9 does not compute the discretionary data, which is dynamic in nature (e.g. the discretionary  
10 data is dynamic data). In an example embodiment, the expiry date of the virtual card is  
11 identical to the expiry date of the funding card, so that from the perspective of the merchant  
12 and the user the virtual card is identical to the funding card. In fact, based on certain  
13 similarities between the funding card and the virtual card, the merchant and the user will be  
14 unaware that a virtual card is being generated and used and will instead believe that the  
15 funding card is being used in the payment. The virtual card details may further include an  
16 internal expiry date that is known only to the payment gateway server, and has a short  
17 timeline of about a few days from the date that the virtual card is created. The internal expiry  
18 date is different from the virtual card's expiry date, and the function of the internal expiry date  
19 is to provide an additional indicator to the payment gateway to determine whether or not a  
20 virtual card has expired. The payment gateway server 506 encrypts the virtual card data,  
21 which does not include the internal expiry date, and sends the encrypted virtual card payload  
22 to the mobile device's payment application 620 (block 711).

23 **[0095]** As an alternate example embodiment, instead of the payment gateway server  
24 506 sending the virtual card PAN as part of the encrypted virtual card payload to the mobile  
25 device, the payment gateway server 506 instead sends a key value (called Kpan) that the  
26 mobile device can use to generate an identical virtual card PAN as computed by the  
27 payment gateway server 506.

28 **[0096]** In an example embodiment, the last four digits of the virtual card PAN are the  
29 same as the last four digits of the funding card PAN. The last four digits of the virtual card  
30 PAN are identical to the funding card PAN so that, after PAN truncation, it appears as if the  
31 PAN of the virtual card is identical to the PAN of the funding card. In other words, the user  
32 cannot detect that a virtual card is being used in place of a funding card. By way of  
33 background, PAN truncation is enforced by the payment industry as part of the merchant  
34 certification process for accepting card payment. A truncated PAN means that the card

1 number, when printed on a customer receipt, is replaced with a printout of only the last four  
2 digits, and the remainder of the other PAN digits are replaced usually by asterisks. An  
3 example embodiment of a truncated PAN is \* \* \* \* \* \* \* \* \* 7777. This hides the card  
4 number from anyone who obtains the receipt when discarded, or by other means, while still  
5 allowing a cardholder with multiple cards to identify which was used, and thus record the  
6 transaction.

7 **[0097]** In an example embodiment, the first portion of digits of the PAN of the virtual card  
8 is static and refers to the payment gateway server 506. For example, the first six digits point  
9 to the payment gateway server 506; the merchant acquirer 103 and associated payment  
10 network uses this information to send the transaction and payment details to the payment  
11 gateway server 506.

12 **[0098]** In an example embodiment, the PAN of the virtual card is nineteen digits long  
13 and compliant to the algorithm LUHN-10. The algorithm, also known as the "modulus 10" or  
14 "mod 10" algorithm, is a checksum formula used to validate a variety of identification  
15 numbers, such as card numbers. As described above, the first six digits are used to identify  
16 the payment gateway server and the last four digits are identical to the last four digits of the  
17 funding card PAN. The remaining digits can be computed in a number of ways. In an  
18 example embodiment, the remaining digits of the virtual card PAN are randomly generated.  
19 In another example embodiment, the remaining digits are computed using the Kpan value;  
20 further details in this regard are described in Fig. 11. Other methods can be used to  
21 compute the virtual card PAN.

22 **[0099]** The mobile device 501 receives the encrypted virtual card payload, decrypts the  
23 encrypted communication and extracts the virtual card details (e.g. the virtual card PAN and  
24 other card details).

25 **[00100]** In another example embodiment, if the virtual card payload includes a Kpan (e.g.  
26 a key value) instead of a virtual card PAN, the mobile device 501 uses the Kpan to compute  
27 the virtual card PAN.

28 **[00101]** The payment application 620 on the mobile device 501 displays a GUI  
29 requesting the user 101 to input their PIN (e.g. which should be the same PIN provided  
30 when the user registered for the service) (block 712). The application 620 receives the PIN  
31 from the user 101 (e.g. the user enters in the PIN) (block 713). The application 620 uses the  
32 PIN to compute the discretionary data of the virtual card (block 714). With the discretionary  
33 data computed, the Track Two data set is complete. The virtual card data set (e.g. the Track

1 Two data) is delivered to the mobile device's secure element 622 (block 715). In an  
2 example embodiment, data to be sent over the NFC subsystem 609 needs to be provided by  
3 the secure element 622 (block 716). In another example embodiment, though not shown,  
4 the operating system of the mobile device 501 can directly transfer information to the NFC  
5 subsystem 609 without the use of the secure element. The virtual card data set includes the  
6 virtual card PAN, the expiry date, the discretionary data and all other elements in a Track  
7 Two data set to complete a standard contactless payment transaction. The mobile's  
8 payment application 620 displays a message to the user to "Tap to pay" (block 717).

9 **[00102]** At this stage, the user 101 touches or positions the mobile device 501 in close  
10 proximity to the POS terminal device 502 (block 718). When the mobile device 501  
11 becomes in communication range to the POS terminal device 502, the mobile device 501  
12 sends the virtual card data to the POS terminal device 502, using the mobile device's NFC  
13 subsystem 609 (block 719) following the payment network standard for such contactless  
14 payment transaction. The transfer of data occur using radio communication means. In an  
15 example embodiment, the POS terminal device 502 displays a message to the user 101 to  
16 "Remove card" (block 720) once the terminal as received all the card data it needed. The  
17 user positions the mobile device 501 away from the POS terminal device (block 721). Other  
18 computations and processes can occur within the mobile device 501 which completes the  
19 mobile device's participation in the transaction process (block 722).

20 **[00103]** Although not shown in Fig. 7, the POS terminal device 502 will send the virtual  
21 card details to the merchant acquirer, along with other transaction data (e.g. cost for  
22 transaction, merchant ID, etc.). This information is eventually routed to the payment  
23 gateway server 506, as identified by the first portion of digits of the virtual card's PAN. See  
24 Fig. 8 for more details.

25 **[00104]** In an example embodiment, one PIN can be used for all the funding cards  
26 associated with the mobile device. In another example embodiment, the registration process  
27 can request a specific PIN for each funding card. In other words, if a user selects a different  
28 funding card, the user will need to enter in a different PIN.

29 **[00105]** In a preferred example embodiment, the mobile device's payment application 620  
30 does not verify the PIN. Instead, the PIN is indirectly or implicitly verified by the payment  
31 gateway server 506 when verifying the virtual card data set. In other words, the payment  
32 gateway server 506 uses the PIN that was stored at registration time to compute the virtual  
33 card data set. If an incorrect PIN was provided by the user or an adversary during the  
34 transaction, it will cause the computed virtual data to result in a different, incorrect, value

1 compared to the virtual card data computed by the payment gateway server. Once the  
2 received virtual card data set is compared with the expected value by the server, an invalid  
3 PIN entry can be detected. When the payment gateway server receives and verifies the  
4 virtual card data set and detect an un-expected virtual card data set (e.g. due to the incorrect  
5 PIN), then the payment authorization can be declined.

6 **[00106]** Turning to Fig. 8, example computer executable or processor implemented  
7 instructions are provided for facilitating a transaction between a mobile device 501 and a  
8 POS terminal device 502, including further details regarding the processing of data by the  
9 merchant acquirer 103, payment gateway server 506 and funding card issuer server 104.

10 **[00107]** Blocks 701, 718, 719, 720, and 721 shown again to provide context for the  
11 process. After the POS terminal device 502 receives the virtual card data from the mobile  
12 device 501 (block 719), the POS terminal device 502 sends the virtual card data, the  
13 transaction information (e.g. amount of payment), and other information (e.g. merchant ID) to  
14 the merchant acquirer 103 (block 801). In an example embodiment, the data sent by the  
15 POS terminal device 502 is in a standard payment authorization request format.

16 **[00108]** The merchant acquirer 103 sends the payment authorization request, which  
17 includes at least the virtual card data (e.g. virtual card's Track Two data) and the payment  
18 amount, to the payment gateway server 506 (block 802) via the payment network. The  
19 acquirer 103 and the payment network are able to identify the payment gateway server 506  
20 by the first portion of digits in the PAN of the virtual card. It can be appreciated the PAN is  
21 part of the Track Two data.

22 **[00109]** The payment gateway server 506 validates the virtual card data (block 803). To  
23 validate the virtual card data received from the mobile device (via the merchant acquirer  
24 103), the payment gateway server computes the Track Two data set on its own. The Track  
25 Two data computation includes the payment gateway server computing the discretionary  
26 data using the PIN originally received and stored during registration by the user. In an  
27 example embodiment, some of the Track Two data portions (like the PAN and expiry date)  
28 were pre-computed and stored; these pre-computed data portions can be compared against  
29 the received Track Two data. If the virtual card data is successfully validated (received card  
30 Track Two data set matches computed card Track Two data set by the payment gateway  
31 server), then the payment gateway server retrieves the funding card data that is associated  
32 with the virtual card data card(block 804).



1 **[00110]** The validation of the virtual card data in block 803 can also include verifying if the  
2 internal expiry date associated with the virtual card has passed or not. If the present date of  
3 the validation is occurring before or on the internal expiry date, then the virtual card can be  
4 considered validated and the transaction processing can continue. Otherwise, the virtual  
5 card is deemed invalid and the payment authorisation request is denied.

6 **[00111]** If the virtual card is validated, the payment gateway server 506, playing a role  
7 that is similar to a merchant at that point, sends a payment authorisation request to its own  
8 acquirer, where the payment authorisation includes at least the funding card data and the  
9 same payment amount received in block 802, to the funding card issuer server 104 (block  
10 805) via the payment network. The funding card issuer 104 then receives the payment  
11 authorisation request and process the transaction as standard. Once the transaction is  
12 processed, the funding card issuer 104 sends a payment authorisation code to the payment  
13 gateway server (block 806) via the payment network. The authorisation code response  
14 includes whether the payment authorisation has been accepted or denied, a transaction  
15 identification number, etc.

16 **[00112]** The payment gateway server 506 then sends a corresponding payment  
17 authorisation response to the merchant acquirer 103 (block 807) via the payment network,  
18 and the acquirer 103 sends the payment authorisation response to the POS terminal device  
19 502 (block 808) also via the payment network. Although not shown, the payment gateway  
20 server stores the payment authorisation code into the database 509. Although not shown as  
21 well, the POS terminal device 502 may display a message to the end user that the payment  
22 has been accepted or denied, according to the response.

23 **[00113]** In an example embodiment, the payment gateway server 506 may also send a  
24 confirmation to the mobile device 501 that indicates whether or not the payment was  
25 accepted or denied (block 809). The indication can be sent as data specific to the payment  
26 application 620 on the mobile device, or as an email. After receiving such indication, the  
27 mobile device 501 can display a message to the user according to the indication.

28 **[00114]** In an example embodiment, after the payment gateway server 506 receives a  
29 response that the payment has been processed (block 806), the payment gateway server  
30 506 marks the virtual card as being used. In this way, if the same virtual card is used again  
31 in a future transaction, the payment gateway server will decline the transaction. This is  
32 because the virtual card is meant to be used only once.

1 **[00115]** In other example embodiment, the same virtual card can be used for more than  
2 one transaction.

3 **[00116]** Turning to Fig. 9a, an example embodiment of computer executable or processor  
4 implemented instructions is provided for facilitating a transaction using a virtual card. In an  
5 example embodiment, the instructions of Fig. 9a are an example detailed implementation of  
6 blocks 710 and 711. In the example embodiment, the virtual card number (also called the  
7 PAN of the virtual card), the expiry date of the virtual card and the other data are all  
8 computed by the payment gateway server 506. Details explaining the flow of data between  
9 the virtual card issuer server 401, the virtual card operator 402 and the virtual card acquirer  
10 403 are also provided. The entities 401, 402, 403 can be part of the payment gateway  
11 server 506. In another example embodiment, the entities 401, 402, 403 are separate  
12 entities.

13 **[00117]** The operations in blocks 901 through 908 are part of the overall process for the  
14 mobile device getting a virtual card (block 909).

15 **[00118]** The process in Fig. 9a assumes that the mobile device 501 has the payment  
16 application 620 installed, and that the mobile device and the funding cards have been  
17 registered with the payment gateway server 506. It also assumes that a transaction with the  
18 merchant has already been initiated. For example, the POS terminal device 502 has  
19 displayed the message "Tap to pay", and the user has already selected a funding card  
20 through the payment application 620. These conditions are achieved using, for example,  
21 blocks 701, 702, 703, 704, 705, 706 and 707.

22 **[00119]** After receiving a user input to select a funding card, the mobile device 501 sends  
23 the transaction type (selected action in the application menu, in this case, "Pay with a virtual  
24 card") and a funding card identifier. In a preferred example embodiment, the funding card  
25 identifier is determined at the time of registration.

26 **[00120]** The virtual card operator 402 confirms that the selected funding card is valid  
27 (block 902) by ensuring for example that the funding card identifier is linked with the  
28 registered user and that the selected funding card has not expired since its registration.

29 **[00121]** The virtual card operator 402 sends a request to create a virtual card PAN to the  
30 virtual card issuing server 401 (block 903). The request for the virtual card PAN includes the  
31 last four digits of the funding card PAN and the expiry date of the funding card. The virtual  
32 card issuing server creates a virtual card PAN which has the last four digits identical to the  
33 last four digits of the funding card PAN. The virtual card also has the same expiry date and

1 the funding card. The virtual PAN is created by using the first 6 digits assigned to the virtual  
2 card issuer, a unique, never used before, 8 digits long random number (to prevent collision),  
3 and the check mod10 digit.

4 **[00122]** After the virtual card PAN is computed, the virtual card issuing server 401 sends  
5 this information, in a response, to the virtual card operator 402 (block 904). The virtual card  
6 operator will generate portions of the Track Two data such as the PAN, the card expiry,  
7 service code, etc (block 905). In an example embodiment, the discretionary data (which is  
8 part of the Track Two data) is not computed by the payment gateway server 506 at this time.  
9 The virtual card operator 402 associates the virtual card PAN with the funding card PAN  
10 (block 906) and stores it in the database 509 (not shown). It also adds an internal expiry  
11 date to the virtual card records in the database. In an example embodiment, the internal  
12 expiry date is computed to be some predetermined time period after the time that the virtual  
13 card is created. For example, the internal expiry date is 48 hours from the date and time that  
14 the virtual card is created. The virtual card operator 402 then encrypts and sends the virtual  
15 card data (e.g. virtual card PAN, external expiry date, etc.) required by the mobile application  
16 to generate the final virtual card data set card(block 907). The result forms an encrypted  
17 virtual card payload that is sent to the mobile device 501 (block 908).

18 **[00123]** The mobile device 501 receives the PIN from the user (block 925). The mobile  
19 device 501 then uses the PIN to compute the discretionary data (which is a portion of the  
20 virtual card's Track Two data). The computation of the discretionary data can involve a  
21 dynamic value which changes over time, or with each transaction, thereby making the  
22 discretionary data the dynamic data. The data received from the virtual card payload (e.g.  
23 PAN, expiry date, and other data) and the discretionary data form the complete Track Two  
24 data set of the virtual card (block 926).

25 **[00124]** Turning to Fig. 9b, an example embodiment of computer executable or processor  
26 implemented instructions is provided for validating and authorizing a transaction using a  
27 virtual card. The process in Fig. 9b assumes that a virtual card was already created (e.g. as  
28 per Fig. 9a) and focuses on the processing for verifying the virtual card. In an example  
29 embodiment, the instructions of Fig. 9b are an example detailed implementation of blocks  
30 802 to 807. The entities 401, 402, 403 can be part of the payment gateway server 506. In  
31 another example embodiment, the entities 401, 402, 403 are separate entities.

32 **[00125]** The processes described with respect to blocks 910 through 923 are part of the  
33 overall authorization process of the virtual card (block 924).

1 **[00126]** The flow starts after the mobile device 501 has received the encrypted virtual  
2 card payload, decrypted the virtual card payload, captured via the GUI the user's PIN, and  
3 computed the complete virtual card data set (e.g. the complete Track Two data set). At that  
4 point, the mobile application can send the virtual card data set to the POS terminal device  
5 502 via the NFC subsystem 609 (block 910). The POS terminal device 502 sends the virtual  
6 card data, along with transaction data, to the merchant acquirer 103 (block 911). The  
7 merchant acquirer 103 sends a payment authorisation request, which includes the virtual  
8 card data and the amount of payment, to the virtual card issuing server 401 (block 913) via  
9 the payment network.

10 **[00127]** The virtual card issuing server 401 sends the virtual card data to the virtual card  
11 operator 402 (block 914). The virtual card operator 402 uses the virtual card data, such as  
12 the virtual card PAN, to retrieve the associated user profile and funding card data (e.g. PAN  
13 and expiry date) (block 915). The virtual card operator 402 then verifies the received virtual  
14 card data, for example, by comparing data with the server computed version of the expected  
15 card data. The validation process is similar to operation of block 803.

16 **[00128]** If the virtual card data is successfully verified, then the virtual card operator 402  
17 uses the funding card details associated with the virtual card transaction and send a  
18 payment authorisation request, which includes the funding card data and the corresponding  
19 payment amount, to the virtual card acquirer 403. The virtual card acquirer 403 forwards the  
20 details to the funding card issuing server 104 (block 917) via the payment network.

21 **[00129]** The funding card issuing server 104 receives and verifies the funding card data. If  
22 the funding card data is successfully verified, the funding card issuing server sends a  
23 payment authorisation code response to the virtual card acquirer 403, which forwards the  
24 same to the virtual card operator 402 (block 918), all via the payment network. The payment  
25 authorisation code response indicates whether the funding card data is accepted or denied,  
26 and a transaction identification number, etc.

27 **[00130]** The virtual card operator 402 sends a corresponding validation authorisation  
28 code response to the virtual card issuing server 401 (block 919). The virtual card operator  
29 402 and the virtual card issuing server 401 marks the virtual card as being used (block 920  
30 and block 921). In this way, if the same virtual card is used again at a later time, the virtual  
31 card operator 402 or the virtual card issuing server 401 will be able to detect a possible  
32 fraud.

1 **[00131]** The virtual card issuing server 401 sends the payment authorisation code  
2 response for the virtual card to the virtual card operator 402 (block 922). This payment  
3 authorisation code response for the virtual card is then forwarded to the merchant acquirer  
4 103 (block 923) via the payment network.

5 **[00132]** As described earlier, in other example embodiments, the virtual card is not  
6 computed for every transaction. For example, after the virtual card is computed according to  
7 blocks 708, 709, 710, 711, 712, 713, and 714, for one or more subsequent transactions, the  
8 same virtual card can be used for the transaction. In other words, blocks 708, 709, 710,  
9 711, 712, 713, and 714 are not performed for the one or more subsequent transactions.  
10 This saves on processing power and reduces the computation time when making a  
11 transaction. This may also be advantageous when the mobile device 501 is not able to  
12 communicate with the payment gateway server 506. In other words, even if the mobile  
13 device 501 is not able to communicate with the payment gateway server 506, the mobile  
14 device can use the already computed virtual card to perform the transaction.

15 **[00133]** In an example embodiment where the same virtual card is used for subsequent  
16 transactions, certain events, or time periods, or both may trigger a new virtual card to be  
17 computed and loaded on the mobile device 501. In an example embodiment, a new virtual  
18 card is computed when a given number of transactions have been performed using the  
19 previous virtual card, and when the mobile device 501 is able to communicate with the  
20 payment gateway server 506. The given number of transactions is, for example, a randomly  
21 generated number that is generated each time a new virtual card is issued to the mobile  
22 device 501. In another example embodiment, a new virtual card is computed for a  
23 transaction when a given number of days has passed since the previous virtual card was  
24 computed, and when the mobile device 501 is able to communicate with the payment  
25 gateway server 506. The given number of days is, for example, a randomly generated  
26 number that is generated each time a new virtual card is issued to the mobile device 501. It  
27 can be appreciated that randomly generated numbers prevent attackers from predicting  
28 when the next new virtual card will be computed for the mobile device.

29 **[00134]** Turning to Fig. 10, an example embodiment of computer executable or processor  
30 implemented instructions are provided for registering a funding card and mobile device with  
31 payment gateway server. Such an embodiment can be used in combination with the  
32 principles described above. The mobile device 501 sends a registration request to the  
33 payment gateway server 506 (block 1001). The request includes the mobile device's device  
34 ID, a user provided PIN, and funding card details. After receiving the request, the payment

1 gateway server sends a registration response to the mobile device (block 1002). The  
2 response include a certificate, an Application Transaction Counter (ATC), a key value (called  
3 Kpan) for generating a PAN, a secure element key value (called Ksec), and a funding card  
4 identifier that identifies each registered funding card(s). The certificate is a client certificate  
5 for the mobile device 501 and, in an example embodiment, it is configured to according to  
6 the RSA algorithm and has a 2048 bit length. The ATC is a counter that is initially set to a  
7 random value between "0" or "1000" and increments with each transaction. The initialized  
8 ATC value is a random value to prevent adversaries from predicting the ATC values. Copies  
9 of the ATC are stored and synchronise on both the payment gateway server and the mobile  
10 device. In an example embodiment, the ATC is a 10 digit value. In an example  
11 embodiment, the Kpan is 128 bits long, and the Ksec is 128 bits long.

12 **[00135]** The payment gateway server 506 stores the data from the registration request  
13 and the registration response in association with each other (block 1003). The mobile  
14 device 501 also stores the data from the registration response (block 1004).

15 **[00136]** Turning to Fig. 11, an example embodiment of computer executable or processor  
16 implemented instructions are provided for computing virtual card data. Such an embodiment  
17 can be used in combination with the principles described above for computing virtual card  
18 data. The mobile device 501 sends a virtual card request to the payment gateway server  
19 506 (block 1101). This is similar to block 901 in Fig. 9a. The request includes the device ID,  
20 the identifier of the selected funding card, the Certificate and the stored Kpan. The stored  
21 Kpan can be from registration or from the previous transaction.

22 **[00137]** The payment gateway server 506 determines if the received Kpan and Certificate  
23 match the Kpan and Certificate stored on the payment gateway server (block 1102)  
24 associated with the DeviceID stored in the payment gateway server database. If so, then  
25 the payment gateway server 506 sends a virtual card response to the mobile device 501 that  
26 includes a new Kpan (block 1103). A new Kpan is used to generate a different PAN from the  
27 previous transaction, and to also prevent against replay attacks. The payment gateway  
28 server 506 computes the PAN for the virtual card using the new Kpan (block 1104). This  
29 new Kpan and the pre-computed PAN are stored by the payment gateway server 506 for  
30 later use (block 1105).

31 **[00138]** After the mobile device receives the new Kpan, it replaces the stored Kpan with  
32 the new Kpan (block 1106). It uses the new Kpan to compute a PAN for the virtual card, in  
33 the same way the payment gateway server computed the PAN (block 1107). If the  
34 conditions and data are correct, although the mobile device computes the PAN

1 independently of the payment gateway server, the PAN computed by the mobile device  
2 should be identical to the PAN computed by the payment gateway server.

3 **[00139]** The mobile device then computes the discretionary data (which is part of the  
4 Track Two data) using the PIN, the Ksec, the PAN, the Certificate, and the ATC (block  
5 1108). The ATC and the PAN keep changing with each transaction, which causes the  
6 discretionary data to be dynamic data.

7 **[00140]** The mobile device increments the ATC by 1 (block 1109).

8 **[00141]** In particular, the PAN is computed by the mobile device and the payment  
9 gateway server according to the following:

10  $PAN = BIN(6) + SHA256[Kpan](8) + Luhn(1) + Reserved(4)$

11

12 wherein

13 BIN(6) is a 6-digit binary number to identify the payment gateway server;

14 SHA256[Kpan](8) is an 8-digit number generated by taking the sha256 of the Kpan value,  
15 further including converting the sha256 value to decimal and truncating to eight digits from  
16 the hash value;

17 Luhn(1) is a single digit used to ensure the virtual card always passes LUHN algorithm;

18 and

19 Reserved(4) is a 4-digit number that is the same as the last 4 digits of the funding card PAN.

20

21 **[00142]** The above values are concatenated together to form the PAN. The symbol "+" in  
22 the above computation refers to the concatenation operation.

23 **[00143]** In an example embodiment of computing the PAN, the other values may be  
24 known or computed first, and the LUHN value is computed last. For example, an  
25 intermediate form to the PAN is 66666612345678X1111, whereby X represents Luhn(1). In  
26 other words, the BIN(6) = 666666; SHA256[Kpan](8) = 12345678; and Reserved(4) = 1111.

27 **[00144]** Different LUHN algorithms can be used to solve for X, so as to ensure the PAN  
28 satisfies the LUHN algorithm. For example, an example embodiment algorithm includes  
29 (step 1) doubling the value of every other digit, starting from the right-most digit; (step 2)  
30 summing the all the individual digits including the digit X from step 1; and (step 3) solving the

1 expression  $\text{mod}10(\text{sum total from step 2})=0$  for the digit X. In this example embodiment, the  
2 solution to the equation is  $X = 9$ . Other LUHN algorithms can be used.

3 **[00145]** In an example embodiment, any digit before the last 4 digits of the PAN that  
4 satisfies the LUHN criteria can be used.

5 **[00146]** The discretionary data is computed using the following:

6 Discretionary data =  $\text{HMAC\_SHA256}[\text{Ksec}+\text{PIN},\text{M}](10)$

7  
8 where M=concatenation of (PAN,Certificate ID, ATC)

9 **[00147]** Truncation is performed by encoding the SHA results in decimal, then taking the  
10 left most digits. It can be appreciated that SHA256 is a known cryptographic hash function,  
11 and HMAC is a hash based message authentication code involving a cryptographic hash  
12 function in combination with a secret cryptographic key. The secret cryptographic key of the  
13 HMAC function is the concatenated values of Ksec and PIN. The message M is the  
14 concatenated values of PAN, Certificate ID and ATC. The Certificate ID is from the  
15 Certificate.

16 **[00148]** Turning to Fig. 12, an example embodiment of computer executable or processor  
17 implemented instructions are provided for verifying virtual card data, and in particular the  
18 verification by the payment gateway server of the discretionary data (e.g. the dynamic data).  
19 Such an embodiment can be used in combination with the principles described above for  
20 verifying the virtual card data. This can be a continuation of the process described in Fig.  
21 11.

22 **[00149]** The mobile device 501 sends a virtual card to the merchant POS terminal 502,  
23 which then creates and sends a payment authorisation request to the merchant acquirer 103  
24 (block 1201). The payment authorisation request is eventually received by the payment  
25 gateway server 506 (block 1202). The payment authorisation request includes, among other  
26 things, the PAN and the discretionary data of the virtual card, as computed by the mobile  
27 device 501.

28 **[00150]** After receiving the request, the payment gateway server 506 uses the PAN to find  
29 the relevant stored associated data (e.g. Kpan, Certificate, Ksec, funding card identifier,  
30 device ID, and other user data). The relevant stored associated data is identified by the  
31 precomputed PAN, which acts as an index to search into the server database. In other  
32 words, the received PAN is compared with a number of precomputed PANs, and if a match



1 is found with a given precomputed PAN, then the stored data associated with that given  
2 precomputed PAN is considered the relevant stored associated data.

3 **[00151]** The payment gateway server 506 uses the relevant stored associated data to  
4 computes its own discretionary data. This can include using the PIN (received during the  
5 user registration process), the stored Ksec, the Certificate ID from the stored Certificate, and  
6 an incremented value of the ATC (block 1203).

7 **[00152]** The computation of the payment gateway's own discretionary data uses the  
8 following:

9 Discretionary data = HMAC\_SHA256[Ksec+PIN,M](10)

10  
11 where M=concatenation of (PAN,Certificate ID, (ATC+1))

12 **[00153]** The ATC+1 represents the incremented ATC value.

13 **[00154]** The payment gateway server 506 determines if its own discretionary data is equal  
14 to the received discretionary data (block 1204). If they are not equal, the payment gateway  
15 server further increments ATC by "1" and re-computes its own discretionary data (block  
16 1205). The process returns to block 1204 to check if the discretionary data sets are equal.  
17 The process involving block 1205 can be repeated, such so that each time the ATC value is  
18 further incremented by 1. This can be done up to a certain number of times (e.g. 10 times),  
19 after which the transaction process will be stopped.

20 **[00155]** If the range of ATC values (e.g. between ATC+1 and ATC+10) does not generate  
21 an identical discretionary data set, then the verification is unsuccessful. The range is to  
22 account for the possibility that the mobile device's ATC counter may have incremented  
23 without the payment gateway server's knowledge. Therefore, a buffer or range of ATC  
24 values is used.

25 **[00156]** If the payment gateway server's own discretionary data is the same as the  
26 received discretionary data, then the payment gateway server replaces the previous ATC  
27 value with the currently incremented ATC value that is used in the computation of its own  
28 discretionary data. In this way, the ATC value stored on the payment gateway server should  
29 now be equal to the ATC value stored on the mobile device.

30 **[00157]** It is noted that if the PIN used by the mobile device to compute the discretionary  
31 data is incorrectly entered by the user (e.g. is not the same as the PIN provided at  
32 registration), then the discretionary data from the mobile device will not equal the payment

1 gateway server's own discretionary data. This is because its own discretionary data is  
2 computed using the PIN provided at registration. In this way, the PIN provided by the user  
3 into the mobile device during each transaction is implicitly verified by the payment gateway  
4 server.

5 **[00158]** Continuing with Fig. 12, after the data is successfully verified or not, the payment  
6 gateway server eventually respond to the merchant by sending an authorisation response to  
7 the merchant (103, 502) (block 1207). To further clarify, if the discretionary data verification  
8 was not successful (as per block 1205 after a certain of iterations has been reached), then a  
9 negative payment authorisation response is sent to the merchant (block 1207).

10 **[00159]** The methods and systems described herein do not rely on having a software  
11 payment card stored on the secure element on the mobile device to facilitate contactless  
12 payment transaction. This is because the PAN, and the other card data elements of the  
13 funding card are not stored on in the mobile device. Instead the PAN and expiry date of the  
14 funding card is stored on a secure server (e.g. the payment gateway server). Furthermore,  
15 the virtual card data is created on a secure server, and is encrypted for a particular mobile  
16 device. Furthermore, the discretionary data in the Track Two data is a function of the PIN  
17 which re-entered with each payment. The virtual card data is only sent at the time of making  
18 the payment, and can only be used once. In addition, only the server can validate the virtual  
19 card data. In other words, there is nothing for an adversary to access via brute force on the  
20 mobile device without being detected.

21 **[00160]** Furthermore, the virtual card encrypted payload is not reusable. This is because  
22 as soon as the card is used once, it cannot be used again by any other party.

23 **[00161]** It is also noted that, some example embodiments, the one-time virtual card is  
24 created just-in-time, ahead of the payment authorization from merchant.

25 **[00162]** Furthermore, the funding card details are never provided to the merchant, since  
26 the virtual card details are provided instead.

27 **[00163]** The methods and systems described herein are fully compatible with standard 4-  
28 party card payment model and systems. There is no system change required by the issuer  
29 of the funding card, merchant, acquirer or payment network.

30 **[00164]** The methods and systems described herein also allow for the use of a "single  
31 application" to be used to support one or many virtual cards as well as one or many funding  
32 cards. This reduces the required storage on the secure element (and generally storage on

1 the mobile device) so as not to limit the number of cards a customer can load into a secure  
2 element.

3 **[00165]** It can be appreciated that the methods and systems described herein enable the  
4 cardholder (e.g. end user) to register any funding card they want into their NFC-enabled  
5 mobile device, independently of the card issuer having the infrastructure or a commercial  
6 relationship with a particular mobile operator.

7 **[00166]** The methods and systems described herein can be used with any type of funding  
8 card. The funding cards and the mobile device are pre-registered with the payment gateway  
9 server 506. This registration is independent of any particular mobile phone carrier and any  
10 particular funding card issuer. As a result, no commercial agreement and additional  
11 computing infrastructure are required by a mobile phone carrier and a funding card issuer to  
12 facilitate contactless payment using a mobile device. This in turn reduces the cost incurred  
13 by the funding card issuer to issue software payment cards to mobile devices.

14 **[00167]** In an example embodiment, a method performed by a server is provided for  
15 facilitating payment. The method includes: receiving a message from a mobile device  
16 identifying a funding card; searching a database of multiple cards associated with the mobile  
17 device for a funding card number associated with the identified funding card; computing data  
18 for a virtual card, the data comprising a card number and an expiry date; storing the data for  
19 the virtual card number in association with the funding card number; sending the data for the  
20 virtual card to the mobile device; computing the card details using the user PIN as input;  
21 receiving a first payment authorisation request from a merchant acquirer, the request  
22 comprising the data for the virtual card and a requested payment amount; retrieving the  
23 funding card number based on the data for the virtual card; sending a second payment  
24 authorisation request to a funding card issuer, the request comprising the funding card  
25 number and the requested payment amount; receiving a payment authorisation response  
26 from the funding card issuer; and sending the payment authorisation response to the  
27 merchant acquirer.

## 28 **E-commerce and Internet based transactions**

29 **[00168]** In another example embodiment of the proposed systems and methods, the  
30 above described principles of the virtual card is also applied to e-commerce or Internet-  
31 based transactions. In other words, while the above examples include the use of a physical  
32 POS terminal device 502 that interacts with the mobile device 501, the e-commerce or  
33 Internet-based transactions do not use the POS terminal device 502. A mobile device 501

1 can instead communicate with the merchant acquirer 103 or the payment gateway server  
2 506 (e.g. via an e-commerce webpage or payment application). In other word, to execute a  
3 transaction, the mobile device sends the virtual card data to the merchant acquirer 103 or  
4 the payment gateway server 506 (e.g. via an e-commerce webpage or payment application)  
5 using Internet connection.

6 **[00169]** For example, the above operations involving the POS terminal device 502 can be  
7 performed, but using an e-commerce webpage or payment application instead of the POS  
8 terminal device 502. As shown in Fig. 13, a schematic diagram view of the entities involved  
9 in an example embodiment of a payment transaction using a virtual card to facilitate a  
10 payment is provided, which is similar to Fig. 5. However, unlike Fig. 5, which shows a POS  
11 terminal device 502, in Fig. 13, an Internet and e-commerce interface 1301 is shown  
12 interacting with the mobile device 501 and the merchant acquirer 103. In another example  
13 embodiment, in addition, or in the alternative, the mobile device 501 interacts with the  
14 payment network 504 directly through the Internet and e-commerce interface as shown by  
15 connection 1302. It is appreciated that the mobile device 501 does not require an NFC  
16 subsystem 609 to perform transactions using the Internet and e-commerce interface 1301.  
17 In other words, the mobile device 501 may or may not have the NFC subsystem 609, without  
18 affecting the storage and retrieval of information on the secure element 622, and without  
19 affecting an Internet or e-commerce transaction. The mobile device connects to the Internet  
20 and e-commerce interface 1301 through a webpage that can be viewed through an Internet  
21 browser application 619 on the mobile device. In another example embodiment, the mobile  
22 device connects to the Internet and e-commerce interface 1301 through a payment  
23 application GUI.

24 **[00170]** In general, the above examples described with respect to Figs. 7, 8, 9a, 9b, 10,  
25 11 and 12 are also used for e-commerce transactions, but with the POS terminal 502  
26 replaced with the Internet and e-commerce interface 1301 as per Fig. 13.

27 **[00171]** An example graphical user interface (GUI) 1401 as displayed by an e-commerce  
28 webpage or payment application GUI is shown in Fig. 14. This is displayed by the mobile  
29 device 501, for example, when a user has selected a product, an item, or a service to  
30 purchase. The transaction details 1402 are shown, which include, for example, the amount  
31 to be paid by the user and the name of the user. Other information, for example, the product  
32 ID or user ID may be shown. The funding card information 1403 is also shown and, in the  
33 case of a credit card, may show the last four digits of the funding card 1404. In an example  
34 embodiment, a default funding card is displayed, but a different funding card can be selected

1 using the control 1407. In the example of Fig. 14, the mobile device 501 will make a  
2 payment using a virtual card associated with the Visa funding card ending in the digits '4242'  
3 (1404). In another example embodiment, the last four digits are not displayed to prevent  
4 attackers from viewing this information.

5 **[00172]** When the GUI 1401 is displayed, to complete the transaction, the user enters in a  
6 PIN into the field 1405 and selects the "pay now" button 1406. When the GUI 1401 detects  
7 these events, the mobile device 501 sends virtual funding card data to the merchant acquirer  
8 103 or the payment gateway server 506, via the Internet and e-commerce interface 1301.

9 **[00173]** In another example embodiment GUI (not shown), the "pay now" button 1406 is  
10 not displayed. For example, the GUI is able to detect the length of how many characters  
11 were entered into the entry field 1405. After the GUI detects that the required number of  
12 characters have been entered in the entry field 1405, the PIN is automatically submitted.

13 **[00174]** In an example embodiment, the PIN is the card verification value (CVV), card  
14 security value (CSV), card security code (CSC), card code verification (CCV), card  
15 verification code (CVC or CVC2), etc., of the funding card and this value is determined by  
16 the funding card issuer 104. In another example embodiment, the PIN is determined by the  
17 user. In another example embodiment, the PIN is a password. In another example  
18 embodiment, the PIN is a password used by the system "Verified by Visa", which is a  
19 supplemental verification.

20 **[00175]** In another example embodiment, as shown in Fig. 15, another example GUI 1501  
21 is provided and it displays transaction details 1402, funding card information 1403, and a  
22 "pay now" button 1406. In other words, when the GUI 1501 is displayed, the user only  
23 needs to select the "pay now" button 1406 to execute the transaction. This will cause the  
24 mobile device 501 to send virtual funding card data to the merchant acquirer 103 or the  
25 payment gateway server 506, via the Internet and e-commerce interface 1301. It is  
26 appreciated that a PIN is not required using the GUI of Fig. 15.

27 **[00176]** Turning to Fig. 16, an example set of processor implemented instructions are  
28 provided for making a payment using an Internet and e-commerce interface 1301. Fig. 16 is  
29 similar to Fig. 7, so similar operations are not repeated in detail. At block 1601, the Internet  
30 and e-commerce interface invokes the mobile device to display data on an e-commerce GUI  
31 (e.g. 1401 and 1501). The mobile device 501 displays the e-commerce GUI at block 1602.  
32 Optionally, though not necessarily, blocks 703 and 704 are performed by the mobile device  
33 501. Furthermore, the mobile device, via the e-commerce GUI, optionally shows pre-

1 registered funding cards at block 706, and the user selects a funding card 707. In other  
2 embodiments, there is only one funding card, or there is a default funding card that is used,  
3 unless changed by the user. The operations at blocks 708, 709, 710, and 711 are  
4 performed. As described above, in some example embodiments, a previously computed  
5 virtual card can be used and so, a new virtual card does not need to be computed for each  
6 and every transaction.

7 **[00177]** Continuing with Fig. 16, in an example embodiment, the e-commerce GUI  
8 requests PIN authentication (block 712) and the user provides the PIN (713). However, in  
9 other embodiments, providing the PIN is not required. The operations in blocks 714 and 715  
10 are also performed. However, in other embodiments, if the previously computed virtual card  
11 is being used, the blocks 714 and 715 are not performed.

12 **[00178]** In an example embodiment, the user inputs a command, via the e-commerce  
13 GUI, to execute the transaction (block 1603). For example, the user can select the “pay  
14 now” button 1406, or provide some other input that is understood to execute the transaction.

15 **[00179]** The mobile device sends the virtual card data to the Internet and e-commerce  
16 interface (block 1604), and the Internet and e-commerce interface sends the same to the  
17 merchant acquirer 103 or payment gateway server 506 (see Fig. 17). When transaction is  
18 complete (either accepted or denied), a confirmation message is sent via the Internet and e-  
19 commerce interface to the mobile device (block 1605). Block 722 may then be performed.

20 **[00180]** Fig. 17 shows example processor implemented instructions, which is a  
21 continuation from Fig. 16. Blocks 1601, 1603 and 1604 are included to show context, and  
22 were implemented in previous Fig. 16. Fig. 17 is similar to Fig. 8, but instead includes the  
23 Internet and e-commerce interface 1301. The operations of blocks 801, 802, 803, 804, 805,  
24 806, 807, 808 and 809, apply to Fig. 17 and are not described in detail again, since these  
25 blocks were described in detail with respect to Fig. 8.

## 26 **Party-to-party value transfer**

27 **[00181]** In other example embodiments of the proposed systems and methods, the  
28 following relates generally to facilitating a party-to-party value transfer using a virtual card on  
29 a mobile device.

30 **[00182]** Mobile devices can be used to transfer value, for example, between two persons.  
31 A mobile device can be equipped with a near field communication (NFC) system which can  
32 be used to transfer payment credentials, such as payment card information, to another  
33 mobile device that is also equipped with a NFC system.

1 **[00183]** It is recognized that, it is difficult to transfer money from one person's mobile  
2 device to another person's device, or more generally, from one party to another party using  
3 mobile devices. In many cases, mobile phone carriers and funding card issuers do not have  
4 computing systems to support the transfer of money from one mobile device to another  
5 mobile device, or it is cost prohibitive to the parties. It is also recognized that it is difficult to  
6 transfer money between mobile devices in a secure manner, while still maintaining  
7 convenience.

8 **[00184]** It is also recognized that there are situations in which a sending user wishes to  
9 transfer value to a receiving user, but the sending user does not know or trust the receiving  
10 user. For example, a sending user owes money to a receiving user, but does not know or  
11 trust the receiving user. Transferring value (e.g. money) using a funding card to the  
12 receiving user, without providing any data or information about the funding card, can be  
13 difficult. Furthermore, doing so in a quick and convenient way also makes the transfer of  
14 value difficult.

15 **[00185]** It is also recognized that there are situations in which the receiving user does not  
16 have a bank account. In other words, the receiving user does not have an established  
17 account to receive and store the value (or funds) from the sending user. Therefore, in many  
18 cases, the receiving user cannot accept or receive the value (or funds) from the sending  
19 user.

#### 20 **Transferring value party-to-party using a transfer ID**

21 **[00186]** The methods and systems described herein allows a sender (e.g. a first person)  
22 to send value, for example, money, to a receiver (e.g. a second person). In particular, the  
23 sender's NFC-enabled mobile device is "tapped" against the receiver's NFC-enabled mobile  
24 device, and the transfer of value occurs. The transferred value is stored in association with  
25 the receiver's mobile device as a prepaid virtual card. The receiver can then use their NFC-  
26 enabled phone to make payments with the prepaid virtual card, either via NFC (e.g. with a  
27 merchant's NFC-enabled point of sale terminal) or via the Internet (e.g. m-commerce or e-  
28 commerce).

29 **[00187]** As an alternate example embodiment, instead of sending a virtual card, the  
30 payment gateway server 506 sends a transfer ID to the giving user to be shared with the  
31 receiving user. The transfer ID is used by the payment gateway server to identify the  
32 selected funding card and amount specified by the sending user. In other words, the  
33 transfer ID is considered a pointer that points to the information stored on the payment

1 gateway server to identify the selected funding card and amount specified by the sending  
2 user. Non-limiting examples of a transfer ID can be numerals, a collection of characters  
3 (including numerals), and a URL.

4 **[00188]** The systems and methods described herein also allow a cloud-based wallet  
5 payment gateway server to synchronize with a receiver's NFC-enabled mobile device and  
6 application to facilitate contactless transfers of value from a sender's NFC enabled mobile  
7 device. A sender user selects a funding card and provides the amount for making the  
8 contactless party-to-party value transfer, through their mobile device. A transfer ID is  
9 generated by the payment gateway server to complete the party-to-party payment  
10 transaction. The transfer ID is used by the payment gateway server to identify the selected  
11 funding card and the amount to be transferred by the sending user. On the payment  
12 gateway server, the transfer ID is temporarily associated with the funding card and the  
13 allowed amount of value to be transferred.

14 **[00189]** When a party-to-party value transfer is initiated, the data including the transfer ID  
15 data is sent through the NFC system on the sender's mobile device to the receiver's NFC-  
16 enabled mobile device. This information is sent from the receiver's mobile device to the  
17 cloud-based wallet payment gateway server (also referred to the "payment gateway server").  
18 The cloud-based wallet payment gateway server verifies the transfer ID. If successfully  
19 verified, the sending user's funding card details associated with the transfer ID are retrieved  
20 and sent to the funding card issuer server via the payment gateway server to complete the  
21 value transfer authorisation. The funding card issuer server verifies the funding card and  
22 sends back an authorization code to the payment gateway server. The payment gateway  
23 server then acts as a virtual card issuer server and generates a prepaid virtual card for the  
24 receiver user, which can be used by the receiver's mobile device.

25 **[00190]** The value transfer for the prepaid virtual card can be settled when the payment  
26 gateway server initiates a settlement request, typically once at the end of every business  
27 day, using the standard method. This includes retrieving all the funding card numbers and  
28 the corresponding authorization codes received during the period and sending this  
29 information to the funding card issuer for settlement. The funding card issuer verifies the  
30 funding card numbers and authorisation codes, and if successfully verified, sends the money  
31 back to the bank account associated with the payment gateway server via a standard  
32 electronic funds transfer method. The money is stored in association with the prepaid virtual  
33 card account. In an example embodiment, the virtual card issuer is the payment gateway  
34 server, or a module within the payment gateway server.



1 **[00191]** In another example embodiment, before the sender's mobile device sends the  
2 transfer ID to the receiver's mobile device, the sender's mobile device requests authorisation  
3 to the funding card issuer via the payment gateway server. After authorisation is completed  
4 a transfer ID is created and then sent to the receiver's mobile device. The receiver's mobile  
5 device verifies the transfer ID with the payment gateway server. The payment gateway  
6 server then performs a settlement request associated with the authorization request from the  
7 transfer ID. The payment gateway server then generates a prepaid virtual card for the  
8 receiver user, which can be used by the receiver's mobile device.

9 **[00192]** In another example embodiment, before the sender's mobile device sends the  
10 transfer ID to the receiver's mobile device, both the authorisation and the settlement occur.  
11 After the funding card issuer server completes the authorisation, and after the settlement for  
12 the value transfer has been performed and a prepaid virtual card has been created, then the  
13 transfer ID is transferred to the receiver's mobile device.

14 **[00193]** In an example embodiment, the systems and methods described herein allow an  
15 NFC-enabled device with the party-to-party transfer application to transfer a user-defined  
16 amount of value to another NFC-enabled device, also having the party-to-party transfer  
17 application, without ever disclosing the sender's funding card details to the receiver.  
18 Furthermore, the sender is able to set or determine the value to give to the receiver. A  
19 successful transfer results in the receiving user being issued a prepaid virtual card for the  
20 same amount of value as sent by the sending user. Therefore, even if the receiving user  
21 does not have a bank account to receive the value, the receiving user can still be given a  
22 prepaid virtual card which can be used by the receiving user to make payments and  
23 transactions.

24 **[00194]** It can be appreciated that any funding card can be used, and is not limited or  
25 dependent on the mobile phone carrier having agreement with the funding card issuer. For  
26 a card to be used, it first needs to be registered with the service. It can also be appreciated  
27 that any number of funding cards can be registered in association with the mobile device.  
28 The cardholder's mobile device includes a payment application (also called value transfer  
29 application) that can interact with the payment gateway server.

30 **[00195]** In an example embodiment of the registration, for each funding card the sending  
31 user wishes to register, the user types in card details into the mobile device (e.g. the name  
32 printed on the funding card, the PAN printed on the funding card, the expiry date printed on  
33 the funding card, and the static security code printed on the funding card). As mentioned  
34 above, funding card issuer does not need to have an existing agreement with any mobile

1 phone carrier. The mobile device sends this data and a mobile device ID to the payment  
2 gateway server. The payment gateway server computes a funding card identifier which  
3 identifies the given funding card. The payment gateway server stores the funding card  
4 identifier in association with the funding card details, and it sends the funding card identifier  
5 to the mobile device for storage. In another example embodiment, the user simply taps a  
6 taps a contactless card on the mobile device so that the mobile application can capture the  
7 card details and send it to the payment gateway server for registration. In an example  
8 embodiment, the funding card identifier is a value that is different from the PAN, expiry date  
9 or static security code of the funding card. For example, the funding card identifier is a  
10 random value so that, if intercepted by an adversary, would not be able to recognize any  
11 funding card details. In an example embodiment, the mobile device does not store any  
12 funding card details or stores limited funding card details (e.g. the name funding card issuer  
13 and the last 4 digits of the PAN). The mobile device stores the funding card identifier, which  
14 it sends to the payment gateway server to indicate a specific funding card. It can be  
15 appreciated that there are other methods to capture the funding credit card details (e.g.  
16 besides the user typing in the data), which can be used with the principles described herein.

17 **[00196]** It can be appreciated that a single payment application is required on the mobile  
18 device, which can manage multiple funding cards. If multiple funding cards are registered,  
19 each of the associated funding card identifiers are stored on the mobile device, within the  
20 single payment application. The details of each individual funding card are stored on the  
21 payment gateway server. In this way, the payment gateway server acts as a cloud-based  
22 server that stores the details of multiple funding cards.

23 **[00197]** Turning to Fig. 18, an example of computer executable or processor implemented  
24 instructions are provided for facilitating a transfer of value from a sending user 101 to a  
25 receiving user 1807. At block 1801, the sending user specifies a funding card and an  
26 amount to transfer to the receiving user. At block 1802, the sending user receives, from the  
27 payment gateway server, a transfer ID that is based on the amount and the funding card. At  
28 block 1803, the sending user sends the transfer ID to the receiving user. At block 1804, the  
29 receiving user validates the transfer ID and the amount, for example, through the payment  
30 gateway server. At block 1805, if valid, the receiving user is issued a prepaid virtual card for  
31 the amount specified by the sending user.

32 **[00198]** With respect to block 1803, although many of the example embodiments  
33 described herein use NFC technology to transmit data between the mobile devices, it can be  
34 appreciated that other data transmission methods can be used. For example, the transfer ID

1 can be sent to the receiving user's mobile device via Bluetooth, infrared, and other peer-to-  
2 peer (P2P) communication technologies. In other example embodiments, the transfer ID  
3 may be transmitted through other means that may not necessarily be P2P, including instant  
4 messaging, text messaging, barcodes, 2D barcodes, QR code, email, etc. In an example  
5 embodiment, additional data or alternative data is transmitted from the sending user's mobile  
6 device to the receiving user's mobile device in order to facilitate the transfer of value,  
7 resulting in a prepaid virtual card available for use by the receiving user. In an example  
8 embodiment, the transfer ID is securely transmitted, for example, when it is transferred  
9 between any of the server, the sending user's mobile device, and the receiver's user's mobile  
10 device.

11 **[00199]** Turning to Fig. 19, example embodiment components of a system for facilitating a  
12 party-to-party transfer of funds is shown using the transfer ID. The sending user 101 is  
13 shown. The user has one or more funding cards 505. For example, the user has multiple  
14 funding cards. The user 101 also owns an NFC-enabled mobile device 501, which includes  
15 a payment application (also called a value transfer application). The sender's mobile device  
16 1901 is configured to interact, via NFC, with the receiver's NFC-enabled mobile device 1901.  
17 It is appreciated that the receiver's mobile device 1901 may have similar hardware and  
18 software components to the sender's mobile device, as shown in Fig. 6. Both the sender's  
19 mobile device 501 and the receiver's mobile device 1901 interact with the payment gateway  
20 sever 506. The payment gateway server 506 is also in data communication with a funding  
21 card issuing server 104.

22 **[00200]** The payment gateway server 506 stores data components specific to the sending  
23 user (block 1903) and data components specific to the receiving user (block 1902). For  
24 example, data components specific to the sending user (block 1903) include registered data  
25 associations 1904 that associate one or more funding cards with the sender's mobile device  
26 ID. For example, funding card1 and funding card2 (and other funding cards) are stored in  
27 association with the mobile device ID of the sender's mobile device 501. There are also  
28 temporary data associations 1905, which includes a transfer ID 1907 being temporarily  
29 associated with one of the user's funding cards 1906. The transfer ID 1907 can be used by  
30 the sending user 101 to make a party-to-party value transfer.

31 **[00201]** The data components related to the receiving user (block 1902) include  
32 registered data associations 1908 that specify zero or more funding cards are associated  
33 with the receiver's mobile device ID. For example, the funding cards associated with the  
34 receiver's mobile device ID can include two funding cards and the prepaid virtual card 1909

1 issued by the payment gateway server 506. It can be appreciated that the prepaid virtual  
2 card 1909 for the receiving user 1807 can be used to make a payment to a merchant or to  
3 yet another party-to-party value transfer to another user.

4 **[00202]** The payment gateway server 506 is in communication with the mobile devices  
5 501, 1901 through a wireless network. For example, the wireless network is provided by a  
6 mobile phone carrier. The payment gateway server 506 is in data communication with the  
7 funding card issuing server 104 through wired or wireless means, or both.

8 **[00203]** Turning to Figs. 20, 21 and 22, a sending mobile device 501 and a receiving  
9 mobile device 1901 are shown interacting with each other when sending funds via NFC.

10 **[00204]** Turning to Fig. 20, the receiver's mobile device 1901 shows an example  
11 embodiment of a graphical user interface (GUI) that allows a user to send or receive funds.  
12 Although not shown, the sender's mobile device 501 could have displayed a similar GUI and  
13 then, based on a user selection to send funds, the mobile device 501 is placed in a mode to  
14 send funds. In "send funds" mode, the sender's mobile device 501 shows a GUI for  
15 facilitating the sending of funds. The GUI displayed in Fig. 20 on the sender's mobile device  
16 501 is an example embodiment of such a GUI for facilitating the sending of funds.

17 **[00205]** Continuing with Fig. 20, the GUI on the sender's mobile device 501 includes an  
18 input field 2001 that allows the sender to specify how much money to send in a party-to-  
19 party transfer. For example, \$100 can be specified. It also includes a menu of funding cards  
20 that can be selected to make the party-to-party transfer. There is an option, for example, to  
21 use the Visa credit card from CIBC (e.g. a name of a first issuing bank) 2002; there is  
22 another option to use the Mastercard credit card from the Bank of Montreal or BMO (e.g. a  
23 name of a second issuing bank) 2003. The user selects 2004 the Visa credit card 2002 as  
24 the funding card.

25 **[00206]** The receiver's mobile device 1901 also has a payment application that displays a  
26 GUI with a control to receive funds 2006 and a control to send funds 2005. The receiving  
27 user provides a selection input 2007, by selecting the control 2006, which places the mobile  
28 device 1091 in a mode to receive funds.

29 **[00207]** Turning to Fig. 21, after the required information has been entered into the  
30 payment applications of the respective mobile devices 501, 1901, the funds are ready to be  
31 transferred. The sender's mobile device 501 displays a message to "Tap phones to send  
32 funds of \$100 using CIBC Visa". The receiver's mobile device 1901 displays a message to

1 “Tap phones to receive funds”. At this stage, the two mobile devices 501, 1901 are placed  
2 close enough together (e.g. tapped) to allow the successful transfer of data via NFC.

3 **[00208]** Turning to Fig. 22, after the money has transferred from the sender to the  
4 receiver, the sender’s mobile device 501 displays a message that “You have sent \$100 to  
5 Bob (647-667-1234)”. The message, for example, includes the amount transferred, the  
6 name of the receiving user, and a phone number of the receiver’s mobile device. The  
7 receiver’s mobile device 1901 displays a message that “You have received a pre-paid virtual  
8 card of \$100 from Alice (416-333-4321)”. The message, for example, includes the amount  
9 received, the name of the sending user, and a phone number of the sender’s mobile device.

10 **[00209]** Turning to Fig. 23, example computer executable or processor implemented  
11 instructions are provided for facilitating a party-to-party transfer of funds between a sending  
12 user and a receiving user.

13 **[00210]** In an example embodiment, it is assumed that both the sending and the receiving  
14 users each have a NFC-enabled device. It is also assumed that both mobile devices have  
15 installed a payment application. It is also assumed that both users have registered their  
16 mobile devices with a payment gateway server. It is also assumed that the sending user has  
17 also pre-registered one or more funding cards with the payment gateway server, which are  
18 each identified by a funding card identifier. In other words, the payment gateway server has  
19 funding card details stored in association with the sender’s mobile device ID.

20 **[00211]** The sending user 101 starts the payment application 620 on their mobile device  
21 501 (block 2301). The payment application 620 determines if the user has successfully  
22 registered to the service (block 2302), and if so, an application menu is shown, and displays  
23 the option to give using the payment application (block 2303). The user 101 selects the  
24 option to give money using the payment application. It is noted that various user  
25 experiences can be used with the principles described herein.

26 **[00212]** For example, a GUI on the menu is able to receive an input from the user to  
27 initiate a party-to-party transfer of funds with a virtual card with the payment application.  
28 Examples of other menu items include “add a funding card”, “delete a funding card”, etc.

29 **[00213]** The mobile device 501 then displays the funding cards that have been pre-  
30 registered by the user (block 2305). A user input is received to specify an amount to be  
31 transferred (e.g. given) and an input is received to select one of the funding cards (block  
32 2306). In an example embodiment, the displayed funding card information is loaded after  
33 the end user successfully registers to the service and has registered at least one funding

1 card. The list is updated when the user adds an additional funding card into the payment  
2 application or when a funding card is deleted. For each registered funding card, there is a  
3 corresponding record stored on the payment application 620 and on the payment gateway  
4 server database 506 which includes an identifier for the payment network associated with  
5 the funding card, a funding card identifier, etc.

6 **[00214]** The payment application 620 sends the mobile device identifier to the payment  
7 gateway server 506 for device authentication (block 2307). The payment application 620  
8 also sends the transaction type (selected action in the application menu, in this case, "Make  
9 a party-to-party transfer"), the amount to be transferred (e.g. given) and the funding card  
10 identifier for the selected funding card to the payment gateway server 506 (block 2308). The  
11 payment gateway server 506 stores the information sent by mobile device 501, and it also  
12 computes the details regarding a transfer ID (block 2309). The transfer ID is associated with  
13 the funding card and the specified amount. An expiry date may also be associated with the  
14 transfer ID, such that the transfer ID is no longer valid after a certain period of time. For  
15 example, after a few minutes, or a few hours, or a day, or some other time period starting  
16 from the creation of the transfer ID, the transfer ID is no longer valid. In other words, if the  
17 sending user is to transfer value to the receiving user, it is to be done before the time period  
18 expires. The payment gateway server 506 sends the transfer ID to the sending user's  
19 mobile device 501 (block 2310). In an example embodiment, the transfer ID is encrypted  
20 and is able to be decrypted by the sending user's mobile device application 620.

21 **[00215]** As an alternate example embodiment, instead of the payment gateway server  
22 506 sending the transfer ID to the mobile device, the payment gateway server 506 instead  
23 sends a key value that the mobile device can use to generate an identical transfer ID as  
24 generated by the payment gateway server 506. Sending a key value may be safer, should  
25 an attacker intercept the key value. It is appreciated that the mobile device 501 eventually  
26 obtains the transfer ID.

27 **[00216]** The mobile's payment application 620 then displays a message requesting the  
28 user to tap the sender's mobile device 501 with the receiver's mobile device 1901 (block  
29 2311). The sending user 101 tells the receiving user 1807 to get ready to receive the funds,  
30 for example, by "tapping" mobile devices together (block 2312). In other example  
31 embodiments, in addition to or in the alternative, the sending user sends the transfer ID to  
32 the receiving user through other communication means, including Bluetooth, infrared, email,  
33 instant messaging, text messaging, and other wireless and wired means.

1 **[00217]** In an example embodiment, before transferring the transfer ID to the payment  
2 application 620 on the mobile device 501 displays a GUI requesting the sending user to  
3 input their PIN (e.g. which should be the same PIN provided when the sending user  
4 registered for the service). In an example embodiment, the PIN is verified first. In another  
5 example embodiment, the payment gateway server 506 instead sends a key value to  
6 generate a transfer ID, the PIN provided again by the sending user is used with the key  
7 value to generate a transfer ID. If the PIN provided by the sending user is correct, then the  
8 transfer ID generated by the mobile device 501 should be identical to the transfer ID  
9 generated by the payment gateway server 506.

10 **[00218]** Turning to Fig. 24, after the sending user tells the receiving user to get ready, the  
11 receiving user starts the payment application on the receiver's mobile device 1901 (block  
12 2401). The receiver's mobile device 1901 determines if the user has successfully registered  
13 to the service (block 2402), and if so, shows a menu (block 2403). The receiving user  
14 provides an input to initiate the party-to-party receipt of funds using the payment application  
15 (block 2404). Then both the receiving user 1807 and the sending user 101 tap their mobile  
16 devices 501, 1901 together (blocks 2405 and 2406).

17 **[00219]** This tapping (or more specifically the close proximity of the mobile devices)  
18 triggers the sender's mobile device 501 to send the transfer ID, via NFC (block 2407). In  
19 another example embodiment, the transfer ID is transferred to the receiver's mobile device  
20 1901 through other ways (e.g. barcodes, 2D barcodes, QR code, messaging, email, etc.). In  
21 an example embodiment, the transfer ID is sent from the mobile device's secure element  
22 622. In another example embodiment, the sending of the transfer ID does not involve the  
23 mobile device's secure element 622.

24 **[00220]** Continuing with Fig. 24, after the receiver's mobile device 1901 receives the  
25 transfer ID from the sender's mobile device 501, the mobile device 1901 sends the same to  
26 the payment gateway server 506 (block 2408). The payment gateway 506 validates the  
27 transfer ID (block 2409), and if the transfer ID is successfully validated, then the payment  
28 gateway server 506 retrieves the sending user's funding card details and the specified  
29 amount associated with the transfer ID (block 2410).

30 **[00221]** To validate the transfer ID, the payment gateway server may confirm if the  
31 transfer ID itself is correct (e.g. it matches a transfer ID stored on the payment gateway  
32 server). It may also check to see if the transfer ID has expired, and if so, will decline the  
33 transfer. It may also check to see if the transfer ID has been previously used, and if so, will  
34 decline the transfer.

1 **[00222]** The payment gateway server 506 then generates a value transfer payment  
2 authorisation request, which includes the funding card details and the amount to be  
3 transferred to the receiving user 1807 (block 2411). Although not shown, the payment  
4 gateway server 506 then sends this standard payment authorisation request to the funding  
5 card issuing server 104. The funding card issuing server 104 verifies the payment  
6 authorisation request (e.g. verifies enough funds are available, verifies funding card data,  
7 etc.) and provides a payment authorisation response to the payment gateway server 506.  
8 The response indicates if the value transfer is accepted or denied.

9 **[00223]** Assuming the payment using the funding card is accepted, the payment gateway  
10 server 506 marks the transfer ID as used (block 2412). It is noted that the transfer ID is a  
11 one-time use. In other words, when the sender wishes to make another party-to-party  
12 transfer, a new (e.g. different) transfer ID will be created for the other party-to-party transfer,  
13 even if the same funding card and amount are being used. In this way, there is increased  
14 security, and the receiving user will never obtain the details of the sending user's funding  
15 card.

16 **[00224]** The payment gateway server 506 computes a prepaid virtual card to be issued to  
17 the receiving user, and the details associated therewith (block 2413). The prepaid virtual  
18 card is stored in association with the receiver's mobile device ID. The transferred amount of  
19 money (from the sending user) is the amount of money associated (e.g. available) on the  
20 prepaid virtual card. The payment gateway server 506 then sends details about the prepaid  
21 virtual card (e.g. a prepaid virtual card identifier) to the receiver's mobile device (block 2414).

22 **[00225]** In particular, the payment gateway server computes a prepaid virtual card  
23 identifier which identifies the prepaid virtual card account. The payment gateway server  
24 stores the prepaid virtual card identifier in association with the prepaid virtual card details,  
25 and it sends the prepaid virtual card identifier to the receiving user's mobile device for  
26 storage. In an example embodiment, the prepaid virtual card identifier is a value that is  
27 different from the PAN, expiry date or static security code of the prepaid virtual card. For  
28 example, the prepaid virtual card identifier is a random value so that, if intercepted by an  
29 adversary, would not be able to recognize any prepaid virtual card details. In an example  
30 embodiment, the receiving user's mobile device does not store any prepaid virtual card  
31 details or stores limited prepaid virtual card details (e.g. the name of the prepaid virtual card  
32 issuer and the last 4 digits of the PAN). In an example embodiment, the receiving user's  
33 mobile device stores at least the prepaid virtual card identifier, which it sends to the payment  
34 gateway server to indicate the associated prepaid virtual card.



1 [00226] After receipt of the prepaid virtual card identifier, the receiver's mobile device  
2 stores this information (block 2415) and displays a message to the receiving user that the  
3 new prepaid virtual card is available (block 2416).

4 [00227] The payment gateway server 506 may also send a confirmation message to the  
5 sender's mobile device 501 indicating that the new prepaid virtual card has been issued to  
6 the receiver (block 2417).

7 [00228] After the receiver 1807 has the prepaid virtual card, the receiver can use the  
8 prepaid virtual card to make a purchase with a merchant.

#### 9 **Transferring value party-to-party using a virtual card**

10 [00229] In another example embodiment, instead of using a transfer ID, a virtual card is  
11 sent, instead of the transfer ID, to facilitate the transfer of value from the sending user to the  
12 receiving user. The virtual card is associated with the funding card. When using the virtual  
13 card, the receiving user is not able to obtain information about the sending user's funding  
14 card. This increases the security. The results of the transfer is that the receiving user has a  
15 prepaid virtual card that can be used for payment or another party-to-party transfer. The  
16 virtual card is sometimes referred to in the Figures as "Vcard".

17 [00230] By way of background, the secure element is typically managed by a mobile  
18 phone carrier distributing the secure element with the mobile device. It is used in part to  
19 maintain the security of the various applications running inside the secure element. Part of  
20 the managed service includes delivering applications into the secure element directly, or  
21 giving permission to a third party organization to deploy their application on a particular  
22 secure element. The managed service is typically delivered using what the industry referred  
23 to as a trusted service manager (TSM).

24 [00231] All applications stored and running inside the secure element, such as the  
25 individual "software payment card", need their storage own space. Payment cards are  
26 issued to consumers by the card issuer. Deploying software payment cards on mobile  
27 phones requires a high level of coordination between the mobile phone carrier and the card  
28 issuer where the mobile phone carrier provides access to individual secure elements, one at  
29 a time, to the issuer. Only cards from funding card issuers that have the infrastructure and  
30 the agreement with the mobile phone carrier can be delivered and used on the mobile phone  
31 for contactless payment. This is limiting for both the card issuers and for cardholders.

32 [00232] It is also recognized that from the user's perspective, the process of associating  
33 their mobile device with their funding card to be used for contactless payment is very much

1 dependent on pre-arranged relationships between the mobile phone carrier and the card  
2 issuers. Therefore, a user has limited options or no options when determining if their current  
3 funding card can be associated with their mobile phone for contactless payments. For  
4 example, a user has a funding card from Funding Card Issuer A. The user also has a NFC-  
5 enabled mobile phone from Mobile Phone Carrier B. However, Mobile Phone Carrier B only  
6 has a pre-arranged agreement and infrastructure to facilitate contactless payments with  
7 Funding Card Issuer B. Therefore, even if the user wanted to use their mobile phone to  
8 make a contactless payment, the user would not be able to because there is no pre-  
9 arranged agreement and infrastructure between Mobile Phone Carrier B and Funding Card  
10 Issuer A to issue a software payment card into the user's phone. This limits the user's ability  
11 to make NFC-type payments with their mobile device.

12 **[00233]** In another example embodiment, the payment card dynamic data is a rotating  
13 card verification value (CVV, also referred sometime to dynamic CVV or dCVV). This  
14 rotating CVV is computable based on changing information provided by the integrated circuit  
15 inside the card. In another example embodiment, the dynamic data is dynamic EMV data  
16 which is computed using random data from the funding card, or random data from a  
17 merchant's point of sale terminal, or both. A common implementation of dynamic data uses  
18 an Application Transaction Counter (ATC) on the card so that every transaction produces a  
19 different data stream. This is achieved as the ATC is incremented by 1 for every transaction  
20 performed.

21 **[00234]** It is also recognized that a card application specific to a given funding card can  
22 be installed on the mobile device and used to interact with POS terminal as described  
23 above. It is also recognized that the card application is typically installed on the mobile  
24 device's secure element. Typically, each funding card has its own corresponding card  
25 application that resides on the mobile device's secure element. It can be appreciated that as  
26 each card application takes up storage space on the secure element, and that the secure  
27 element typically has very limited storage space, having multiple card applications on the  
28 secure element in some cases is not possible due to insufficient storage space. By way of  
29 background, the secure element can have a native operating system that is programmed to  
30 perform various tasks and activities, including for example, a card application that emulates  
31 the magnetic strip data of a funding card or a card application that emulates the data used in  
32 an EMV contactless payment. Also by way of background, and by way of example, a typical  
33 secure element has memory of 256 kB, and each card application can consume memory of

1 40-80 kB. It can therefore be appreciated that associating multiple funding cards (and each  
2 of their card applications) with a mobile device for NFC payments can be limiting.

3 **[00235]** Therefore, it is desirable to reduce the amount of required storage space card  
4 applications need on the secure element so as to not limit the number of software payment  
5 cards a user can load into a secure element. Along the same lines, it is desirable for mobile  
6 phone carriers to reduce the amount of data used by “card application” on the secure  
7 element so that other types of applications can be loaded thereon. It is also desirable to  
8 reduce costs incurred by the funding card issuer to issue and operate software payment  
9 cards into secure elements. By way of background, a mobile phone carrier typically charges  
10 application providers, such as funding card issuers, for the amount of storage space used on  
11 the secure element. It is also desirable to reduce the amount of infrastructure required by  
12 the funding card issuer to issue a software payment card for the mobile phone. It is also  
13 desirable to reduce the amount of coordination required between the funding card issuer and  
14 the mobile phone carrier to issue a software payment card on particular mobile phone. It is  
15 also desirable to enable the user (e.g. the cardholder) to load any, and as many, funding  
16 cards they want into their NFC-enabled mobile phone, independently of the funding card  
17 issuer having the infrastructure or a commercial relationship or agreement with a particular  
18 mobile phone carrier. It is also desirable to enable the user (e.g. the cardholder) to load any,  
19 and as many, funding cards they want into their NFC-enabled mobile phone, independently  
20 of the funding card issuer having the infrastructure or a commercial relationship or  
21 agreement with a particular mobile phone carrier. It is also desirable to facilitate the transfer  
22 of money between two users in a party-to-party manner using their NFC-enabled mobile  
23 devices.

24 **[00236]** The systems and methods described herein also allow a cloud-based wallet  
25 payment gateway server to synchronize with a receiver’s NFC-enabled mobile device and  
26 application to facilitate contactless transfers of value from a sender’s NFC enabled mobile  
27 device. A sender user selects a funding card and provides the amount for making the  
28 contactless party-to-party value transfer, through their mobile device. A second card, herein  
29 referred to as a virtual card, is generated along with all required card data (e.g. PAN, expiry  
30 date, dynamic data, discretionary data, etc) to facilitate the party-to-party payment  
31 transaction. On the payment gateway server, the virtual card is temporarily associated with  
32 the funding card and the allowed amount of value to be transferred. The sending user sends  
33 the virtual card data and the amount to be transferred to the receiving user (e.g. via NFC or  
34 other communication means). The receiving user sends this data to the payment gateway

1 server, and validates the virtual card data. If validated, the payment gateway server  
2 retrieves the funding card details and uses the same to transfer the specified amount from  
3 the funding card to a new prepaid virtual card. The prepaid virtual card can be used by the  
4 receiving user to make payments, transactions, party-to-party transfers, etc.

5 **[00237]** Turning to Fig. 25, example embodiment components of a system for facilitating a  
6 party-to-party transfer of funds is shown using the virtual card. Fig. 25 is similar to Fig. 19,  
7 however it differs in that the temporary data associations 1905 includes a virtual card (e.g.  
8 virtual card1) 2501 being temporarily associated with one of the user's funding cards 505.  
9 The virtual card 1101 can be used by the sending user 101 to make a party-to-party value  
10 transfer.

11 **[00238]** Turning to Fig. 26 and 27, example computer executable or processor  
12 implemented instructions are provided for facilitating a party-to-party transfer of funds  
13 between a sending user and a receiving user. The examples of Fig. 26 and 27 are similar to  
14 Fig. 23 and 24, but use a virtual card instead of a transfer ID. Therefore, similar elements or  
15 steps are not repeated in the discussion of Fig. 26 and 27.

16 **[00239]** It is assumed that the sender's mobile device has undergone a registration  
17 process, which includes providing PIN. An example of such a process is described later with  
18 respect to Fig. 28.

19 **[00240]** Turning to Fig. 26, blocks 2301 to 2308 are performed. After receiving request  
20 2308, based on the information sent by mobile device 501, the payment gateway server 404  
21 creates and computes the details regarding the virtual card (block 2601). In particular, the  
22 payment gateway server 506 computes a virtual PAN, an expiry date of the virtual card and  
23 other data that forms the Track Two data. It is noted that Track Two data includes, among  
24 other things: the PAN, a service code, an expiry date, discretionary data and a LRC. In an  
25 example embodiment, the payment gateway server 506 at this time does not compute the  
26 discretionary data, which is dynamic in nature (e.g. the discretionary data is dynamic data).  
27 The virtual card details may further include an internal expiry date that is known only to the  
28 payment gateway server, and has a short timeline of about a few days from the date that the  
29 virtual card is created. The internal expiry date is different from the virtual card's expiry date,  
30 and the function of the internal expiry date is to provide an additional indicator to the  
31 payment gateway to determine whether or not a virtual card has expired. The payment  
32 gateway server 506 encrypts the virtual card data, which does not include the internal expiry  
33 date, and sends the encrypted virtual card payload to the mobile device's payment  
34 application 620 (block 2602).

1 **[00241]** As an alternate example embodiment, instead of the payment gateway server  
2 506 sending the virtual card PAN as part of the encrypted virtual card payload to the mobile  
3 device, the payment gateway server 506 instead sends a key value (called Kpan) that the  
4 mobile device can use to generate an identical virtual card PAN as computed by the  
5 payment gateway server 506. This example is described in Fig. 29.

6 **[00242]** In an example embodiment, the first portion of digits of the PAN of the virtual card  
7 is static and refers to the payment gateway server 506. For example, the first six digits point  
8 to the payment gateway server 506; a merchant or payment network, or any other entity, can  
9 use this information to send the transaction and payment details to the payment gateway  
10 server 506.

11 **[00243]** In an example embodiment, the PAN of the virtual card is nineteen digits long  
12 and compliant to the algorithm LUHN-10. The algorithm, also known as the "modulus 10" or  
13 "mod 10" algorithm, is a checksum formula used to validate a variety of identification  
14 numbers, such as card numbers. As described above, the first six digits are used to identify  
15 the payment gateway server. The remaining digits can be computed in a number of ways.  
16 In an example embodiment, the remaining digits of the virtual card PAN are randomly  
17 generated. In another example embodiment, the remaining digits are computed using the  
18 Kpan value; further details in this regard are described with respect to Fig. 29. Other  
19 methods can be used to compute the virtual card PAN. The mobile device 501 receives the  
20 encrypted virtual card payload, it decrypts the encrypted communication and extracts the  
21 virtual card details (e.g. the virtual card PAN and other card details).

22 **[00244]** In another example embodiment, if the virtual card payload includes a Kpan (e.g.  
23 a key value) instead of a virtual card PAN, the mobile device 501 uses the Kpan to compute  
24 the virtual card PAN.

25 **[00245]** Continuing with Fig. 26, the payment application 620 on the mobile device 501  
26 displays a GUI requesting the sending user to input their PIN (e.g. which should be the same  
27 PIN provided when the sending user registered for the service) (block 2603). The  
28 application 620 receives the PIN from the sending user (e.g. the sending user enters in the  
29 PIN) (block 2604). The application 620 uses the PIN to compute the discretionary data of  
30 the virtual card (block 2605). An example embodiment of computing the discretionary data  
31 is described below with respect to Fig. 30. With the discretionary data computed, the Track  
32 Two data set is complete. The mobile's payment application 620 then displays a message  
33 requesting the user to the tap the sender's mobile device 402 with the receiver's mobile

1 device 1901 (block 2311). The sending user 101 tells the receiving user 1807 to get ready  
2 to receive the funds (block 2312).

3 **[00246]** In an example embodiment, one PIN can be used for all the funding cards  
4 associated with the mobile device. In another example embodiment, the registration process  
5 can request a specific PIN for each funding card. In other words, if a user selects a different  
6 funding card, the user will need to enter in a different PIN.

7 **[00247]** In an example embodiment, the mobile device's payment application 620 does  
8 not verify the PIN. Instead, the PIN is indirectly or implicitly verified by the payment gateway  
9 server 506 when verifying the virtual card data set. In other words, the payment gateway  
10 server 506 uses the PIN that was stored at registration time to compute the virtual card data  
11 set, and if an incorrect PIN was provided by the user or an adversary during the transaction,  
12 it will cause the computed virtual data to result in a different, incorrect, value compared to  
13 the virtual card data computed by the payment gateway server. Once the received virtual  
14 card data set is compared with the expected value by the payment gateway server, an  
15 invalid PIN entry can be detected. When the payment gateway server receives and verifies  
16 the virtual card data set and detects an un-expected virtual card data set (e.g. due to the  
17 incorrect PIN), then the value transfer authorization can be declined.

18 **[00248]** Turning to Fig. 27, after the sending user tells the receiving user to get ready, the  
19 receiving user starts the payment application on the receiver's mobile device 1901 (block  
20 2401). Blocks 2401 to 2406 are performed, leading to the mobile devices of both users  
21 tapping their phone together to transfer data via NFC.

22 **[00249]** This tapping (or more specifically the close proximity of the mobile devices)  
23 triggers the sender's mobile device 501 to send the authorised amount and Track Two data  
24 for the virtual card on the mobile device's secure element 622 (block 2701). The mobile  
25 device 501 configures the Track Two data to emulate a virtual card, and this emulated virtual  
26 card data and authorised amount is provided to the mobile device's NFC subsystem 609  
27 (block 2702). The emulated virtual card data includes the complete Track Two data set.  
28 The amount to be transferred and the virtual card data is then transferred, via NFC, to the  
29 receiving user's mobile device 1901 (block 2703). In another example embodiment, though  
30 not shown, the operating system of the mobile device 501 can directly transfer information to  
31 the NFC subsystem 609 without the use of the secure element. The virtual card data set  
32 includes the virtual card PAN, the expiry date, the discretionary data and all other elements  
33 in a Track Two data set to complete a standard contactless value transfer.

1 **[00250]** Continuing with Fig. 27, after the receiver's mobile device 1901 receives the  
2 virtual card data and authorised amount from the sender's mobile device 501, the mobile  
3 device 1901 sends the same to the payment gateway server 506 (block 2704). The  
4 payment gateway 506 validates the virtual card data and authorised amount (block 2705),  
5 and if the data is successfully validated, then the payment gateway server 506 retrieves the  
6 funding card details association with the sender's virtual card (block 2706).

7 **[00251]** To validate the virtual card data received from the sender's mobile device, the  
8 payment gateway server computes the Track Two data set on its own. The Track Two data  
9 computation includes the payment gateway server computing the discretionary data using  
10 the PIN originally received and stored during registration by the sending user. In an example  
11 embodiment, some of the Track Two data portions (like the PAN and expiry date) were pre-  
12 computed and stored; these pre-computed data portions can be compared against the  
13 received Track Two data. If the virtual card data is successfully validated (received card  
14 Track Two data set matches computed card Track Two data set by the payment gateway  
15 server), then the payment gateway server retrieves the funding card data that is associated  
16 with the virtual card data.

17 **[00252]** The validation of the virtual card data in block 2705 can also include verifying if  
18 the internal expiry date associated with the virtual card has passed or not. If the present  
19 date of the validation is occurring before or on the internal expiry date, then the virtual card  
20 can be considered validated and the transaction processing can continue. If validated, the  
21 payment gateway server 506 retrieves funding card details 2706. Otherwise, the virtual card  
22 is deemed invalid and the value transfer authorisation request is denied.

23 **[00253]** When validated, the payment gateway server 506 then generates a value  
24 transfer authorisation payment request, which includes the funding card details and the  
25 amount to be transferred to the receiving user 1807 (block 2707). Although not shown, the  
26 payment gateway server 506 then sends this standard payment authorisation request to the  
27 funding card issuing server 104. The funding card issuing server 104 verifies the payment  
28 authorisation request (e.g. verifies enough funds are available, verifies funding card data,  
29 etc.) and provides a payment authorisation response to the payment gateway server 506.  
30 The response indicates if the value transfer is accepted or denied.

31 **[00254]** Assuming the payment using the funding card is accepted, the payment gateway  
32 server 506 marks the sender's virtual card as being used (block 2708). It is noted that the  
33 virtual card is a one-time use. In other words, when the sender wishes to make another  
34 party-to-party transfer, a new (e.g. different) virtual card will be created for the other party-to-

1 party transfer, even if the same funding card and amount are being used. In this way, there  
2 is increased security, and the receiving user will never obtain the details of the sending  
3 user's funding card.

4 **[00255]** The payment gateway server 506 computes a prepaid virtual card to be issued to  
5 the receiving user, and the details associated therewith (block 2709).

6 **[00256]** After generating the prepaid virtual card, block 2414, 2415, 2416 and 2417 are  
7 performed. These were previously described with respect to Fig. 24.

8 **[00257]** Turning to Fig. 28, an example embodiment of computer executable or processor  
9 implemented instructions are provided for registering a funding card and mobile device with  
10 payment gateway server. Such an embodiment can be used in combination with the  
11 principles described above. The sender's mobile device sends a registration request to the  
12 payment gateway server (block 2801). The request includes the mobile device's device ID,  
13 a user provided PIN, and funding card details. After receiving the request, the payment  
14 gateway server sends a registration response to the sender's mobile device (block 2802).  
15 The response include a certificate, an Application Transaction Counter (ATC), a key value  
16 (called Kpan) for generating a PAN, a secure element key value (called Ksec), and a funding  
17 card identifier that identifies the funding card. The certificate is a client certificate for the  
18 sender's mobile device and, in an example embodiment, it is configured to according to the  
19 RSA algorithm and has a 2048 bit length. The ATC is a counter that is initially set to a  
20 random value between "0" or "1000" and increments with each transaction. The initialized  
21 ATC value is a random value to prevent adversaries from predicting the ATC values. Copies  
22 of the ATC are stored and synchronise on both the payment gateway server and the mobile  
23 device. In an example embodiment, the ATC is a 10 digit value. In an example  
24 embodiment, the Kpan is 128 bits long, and the Ksec is 128 bits long.

25 **[00258]** The payment gateway server 404 stores the data from the registration request  
26 and the registration response in association with each other (block 2803). The sender's  
27 mobile device 402 also stores the data from the registration response (block 2804).

28 **[00259]** Turning to Fig. 29, an example embodiment of computer executable or processor  
29 implemented instructions are provided for computing virtual card data. Such an embodiment  
30 can be used in combination with the principles described above for computing virtual card  
31 data. The sender's mobile device sends a virtual card request to the payment gateway  
32 server (block 2901). The request includes the device ID, the funding card identifier, the



1 Certificate and the stored Kpan. The stored Kpan can be from registration or from the  
2 previous transaction.

3 **[00260]** The payment gateway server determines if the received Kpan and Certificate  
4 match the Kpan and Certificate stored on the payment gateway server (block 2902)  
5 associated with the DeviceID stored in the payment gateway server database. If so, then  
6 the payment gateway server sends a virtual card response to the sender's mobile device  
7 that includes a new Kpan (block 2903). A new Kpan is used to generate a new or different  
8 PAN for each transaction, and to also prevent against replay attacks. The payment gateway  
9 server computes the PAN for the virtual card using the new Kpan (block 2904). This new  
10 Kpan and the pre-computed PAN are stored by the payment gateway server for later use  
11 (block 2905).

12 **[00261]** After the mobile device receives the new Kpan, it replaces the stored Kpan with  
13 the new Kpan (block 2906). It uses the new Kpan to compute a PAN for the virtual card, in  
14 the same way the payment gateway server computed the PAN (block 2907). If the  
15 conditions and data are correct, although the mobile device computes the PAN  
16 independently of the payment gateway server, the PAN computed by the mobile device  
17 should be identical to the PAN computed by the payment gateway server.

18 **[00262]** The mobile device then computes the discretionary data (which is part of the  
19 Track Two data) using the PIN, the Ksec, the PAN, the Certificate, and the ATC (block  
20 2908). The ATC and the PAN keep changing with each transaction, which causes the  
21 discretionary data to be dynamic data.

22 **[00263]** The mobile device increments the ATC by 1 (block 2909).

23 **[00264]** In particular, the PAN is computed by the mobile device and the payment  
24 gateway server according to the following:

25  $PAN = BIN(6) + SHA256[Kpan](12) + Luhn(1)$

26

27 wherein

28 BIN(6) is a 6-digit binary number to identify the payment gateway server;

29 SHA256[Kpan](12) is a 12-digit number generated by taking the sha256 of the Kpan value,  
30 further including converting the sha256 value to decimal and truncating to twelve digits from  
31 the hash value;

32 and

1 Luhn(1) is a single digit used to ensure the virtual card always passes LUHN algorithm.

2 **[00265]** The above values are concatenated together to form the PAN. The symbol “+” in  
3 the above computation refers to the concatenation operation.

4 **[00266]** The discretionary data is computed using the following:

5 Discretionary data = HMAC\_SHA256[Ksec+PIN,M](10)

6 where M=concatenation of (PAN,Certificate ID, ATC)

7 **[00267]** Truncation is performed by encoding the SHA results in decimal, then taking the  
8 left most digits. It can be appreciated that SHA256 is a known cryptographic hash function,  
9 and HMAC is a hash based message authentication code involving a cryptographic hash  
10 function in combination with a secret cryptographic key. The secret cryptographic key of the  
11 HMAC function is the concatenated values of Ksec and PIN. The message M is the  
12 concatenated values of PAN, Certificate ID and ATC. The Certificate ID is from the  
13 Certificate.

14 **[00268]** Turning to Fig. 30, an example embodiment of computer executable or processor  
15 implemented instructions are provided for verifying virtual card data, and in particular the  
16 verification by the payment gateway server of the discretionary data (e.g. the dynamic data).  
17 Such an embodiment can be used in combination with the principles described above for  
18 verifying the virtual card data. This can be a continuation of the process described in Fig.  
19 26.

20 **[00269]** The sender's mobile device 501 sends a virtual card to the receiver's mobile  
21 device 1901 (block 3001). The receiver's mobile device 1901 then creates and sends a  
22 value transfer authorisation request to the payment gateway server 506 (block 3002). The  
23 value transfer authorisation request includes, among other things, the PAN and the  
24 discretionary data of the virtual card, as computed by the sender's mobile device. It also  
25 includes the amount of money or funds to be transferred from the sender to the receiver.

26 **[00270]** After receiving the request, the payment gateway server 506 uses the PAN to find  
27 the relevant stored associated data (e.g. Kpan, Certificate, Ksec, funding card identifier,  
28 device ID, and other user data). The relevant stored associated data is identified by the  
29 precomputed PAN, which acts as an index. In other words, the received PAN is compared  
30 with a number of precomputed PANs, and if a match is found with a given precomputed  
31 PAN, then the stored data associated with that given precomputed PAN is considered the  
32 relevant stored associated data.

1 **[00271]** The payment gateway server 506 uses the relevant stored associated data to  
2 computes its own discretionary data. This can include using the PIN (received during the  
3 user registration process), the stored Ksec, the Certificate ID from the stored Certificate, and  
4 an incremented value of the ATC (block 3003).

5 **[00272]** The computation of the payment gateway's own discretionary data uses the  
6 following:

7 Discretionary data = HMAC\_SHA256[Ksec+PIN,M](10)

8  
9 where M=concatenation of (PAN,Certificate ID, (ATC+1))

10 **[00273]** The ATC+1 represents the incremented ATC value.

11 **[00274]** The payment gateway server 506 determines if its own discretionary data is equal  
12 to the received discretionary data (block 3004). If they are not equal, the payment gateway  
13 server further increments ATC by "1" and re-computes its own discretionary data (block  
14 3005). The process returns to block 3004 to check if the discretionary data sets are equal.  
15 The process involving block 3005 can be repeated, such so that each time the ATC value is  
16 further incremented by 1. This can be done up to a certain number of times (e.g. 10 times),  
17 after which the transaction process will be stopped.

18 **[00275]** If the range of ATC values (e.g. between ATC+1 and ATC+10) does not generate  
19 an identical discretionary data set, then the verification is unsuccessful. The range is to  
20 account for the possibility that the mobile device's ATC counter may have incremented  
21 without the payment gateway server's knowledge. Therefore, a buffer or range of ATC  
22 values is used.

23 **[00276]** If the payment gateway server's own discretionary data is the same as the  
24 received discretionary data, then the payment gateway server replaces the previous ATC  
25 value with the currently incremented ATC value that is used in the computation of its own  
26 discretionary data (block 3006). In this way, the ATC value stored on the payment gateway  
27 server should now be equal to the ATC value stored on the mobile device. This leads to the  
28 authorisation request being accepted and, accordingly, the payment gateway server issues a  
29 virtual prepaid card for the receiver's mobile device (block 1607).

30 **[00277]** It is noted that if the PIN used by the mobile device to compute the discretionary  
31 data is incorrectly entered by the user (e.g. is not the same as the PIN provided at  
32 registration), then the discretionary data from the mobile device will not equal the payment

1 gateway server's own discretionary data. This is because the payment gateway server's  
2 own discretionary data is computed using the PIN provided at registration. In this way, the  
3 PIN provided by the user into the mobile device during each transaction is implicitly verified  
4 by the payment gateway server.

5 **[00278]** Continuing with Fig. 30, after the data is successfully verified or not, the payment  
6 gateway server eventually responds to the receiver's mobile device 1901 by sending a  
7 response to the receiver's mobile device. The response may indicate that the transfer is  
8 declined, or may indicate that a virtual prepaid card has been issued.

9 **[00279]** In particular, as per block 3005, there is an upper limit of iterations that, when  
10 reached, means the discretionary data verification is not successful. If the verification is not  
11 successful, then a declined transfer response is sent to the receiver's mobile device 1901  
12 (block 3008). If the verification is successful, as described above, the process to issue a  
13 virtual prepaid card is started and delivered to the receiver's mobile device 1901 (block  
14 3007).

15 **[00280]** It can therefore be appreciated that a sending user can transfer value to a  
16 receiving user in various ways, including using a transfer ID or a virtual card. The result of  
17 successful transfer is that the receiving user has a prepaid virtual card.

18 **[00281]** In an example embodiment, after the receiving user receives the prepaid virtual  
19 card, the receiving user can use the prepaid virtual card to make a payment with a merchant.  
20 For example, the prepaid virtual card can be used to make a payment using a POS terminal  
21 502 or an e-commerce and Internet base interface 1301.

22 **[00282]** The methods and systems described herein can be used with any type of funding  
23 card. The sender's funding cards and the mobile device are pre-registered with the payment  
24 gateway server 506. This registration is independent of any particular mobile phone carrier  
25 and any particular funding credit card issuer. As a result, no commercial agreement and  
26 additional infrastructure are required between a mobile phone carrier and a funding card  
27 issuer to facilitate contactless (e.g. NFC) payment using a mobile device. This in turn  
28 reduces the cost incurred by the funding card issuer to issue software cards to mobile  
29 devices.

30 **[00283]** In an example embodiment, a method performed by a server for facilitating a  
31 party-to-party value transfer is provided. The method includes: receiving a message from a  
32 sender's mobile device to transfer a specified amount using a funding card identifier;  
33 searching a database of multiple cards associated with the sender's mobile device for

1 funding card details associated with the funding card identifier and amount; generating a  
2 transfer ID and associating the transfer ID with the funding card number and the specified  
3 amount; sending the transfer ID to the sender's mobile device; receiving a value transfer  
4 authorisation request from a receiver's mobile device, the request comprising the transfer ID;  
5 identifying the funding card number and authorised amount based on the transfer ID;  
6 sending a payment authorisation request to a funding card issuer, the request comprising  
7 the funding card number and the specified amount; receiving a payment authorisation  
8 response from the funding card issuer; and sending a value transfer authorisation response  
9 to the receiver's mobile device.

10 **[00284]** In an example aspect, the method further includes: if the value transfer  
11 authorisation response is positive, creating a prepaid virtual card associated with the  
12 receiver's mobile device and having the specified amount.

13 **[00285]** In another example embodiment, a method performed by a server is provided for  
14 facilitating a party-to-party value transfer. The method includes: receiving a message from a  
15 sender's mobile device identifying a funding card and amount; searching a database of  
16 multiple cards associated with the sender's mobile device for funding card details associated  
17 with the identified funding card and amount; computing data for a virtual card, the data  
18 comprising a card number and an expiry date; storing the data for the virtual card number in  
19 association with the funding card number; sending the data for the virtual card to the  
20 sender's mobile device; computing the card details using a PIN as input; receiving a value  
21 transfer authorisation request from a receiver's mobile device, the request comprising the  
22 data for the virtual card and a requested value transfer amount; retrieving the funding card  
23 details based on the data for the virtual card and amount; sending a payment authorisation  
24 request to a funding card issuer, the request comprising the funding card number and the  
25 requested payment amount; receiving a payment authorisation response from the funding  
26 card issuer; and sending a value transfer authorisation response to the receiver's mobile  
27 device.

28 **[00286]** The steps or operations in the flow charts described herein are just for example.  
29 There may be many variations to these steps or operations without departing from the spirit  
30 of the invention. For instance, the steps may be performed in a differing order, or steps may  
31 be added, deleted, or modified.

32 **[00287]** While the basic principles of this invention has been herein illustrated along with  
33 the example embodiments shown, it will be appreciated by those skilled in the art that  
34 variations in the disclosed arrangement, both as to its details and the organization of such

1 details, may be made without departing from the spirit and scope thereof. Accordingly, it is  
2 intended that the foregoing disclosure and the showings made in the drawings will be  
3 considered only as illustrative of the principles of the invention, and not construed in a  
4 limiting sense.

5

6

7

1 **Claims:**

2 1. A computing device configured to facilitate payment, comprising:

3 a processor; and

4 a memory, the memory including processor executable instructions for:

5 computing data for a virtual card, the data comprising a card number and an expiry  
6 date;

7 storing the data for the virtual card in association with a funding card number; and

8 sending the data for the virtual card to a mobile device.

9

10 2. The computing device of claim 1 wherein the processor executable instructions further  
11 comprise:12 receiving a first payment authorisation request from a merchant acquirer, the first  
13 payment authorisation request comprising the data for the virtual card and a requested  
14 payment amount;

15 retrieving information of the funding card based on the data for the virtual card;

16 sending a second payment authorisation request to a funding card issuer, the second  
17 payment authorisation request comprising the funding card number and the requested  
18 payment amount;

19 receiving a payment authorisation response from the funding card issuer; and

20 sending the payment authorisation response to the merchant acquirer.

21

22 3. A mobile device configured to facilitate payment, comprising:

23 a processor; and

24 a memory, the memory comprising processor executable instructions for:

25 receiving an input to make a payment using a funding card;

26 obtaining a virtual card associated with the funding card;

27 sending the virtual card to perform the payment.

28

29

30

31

32

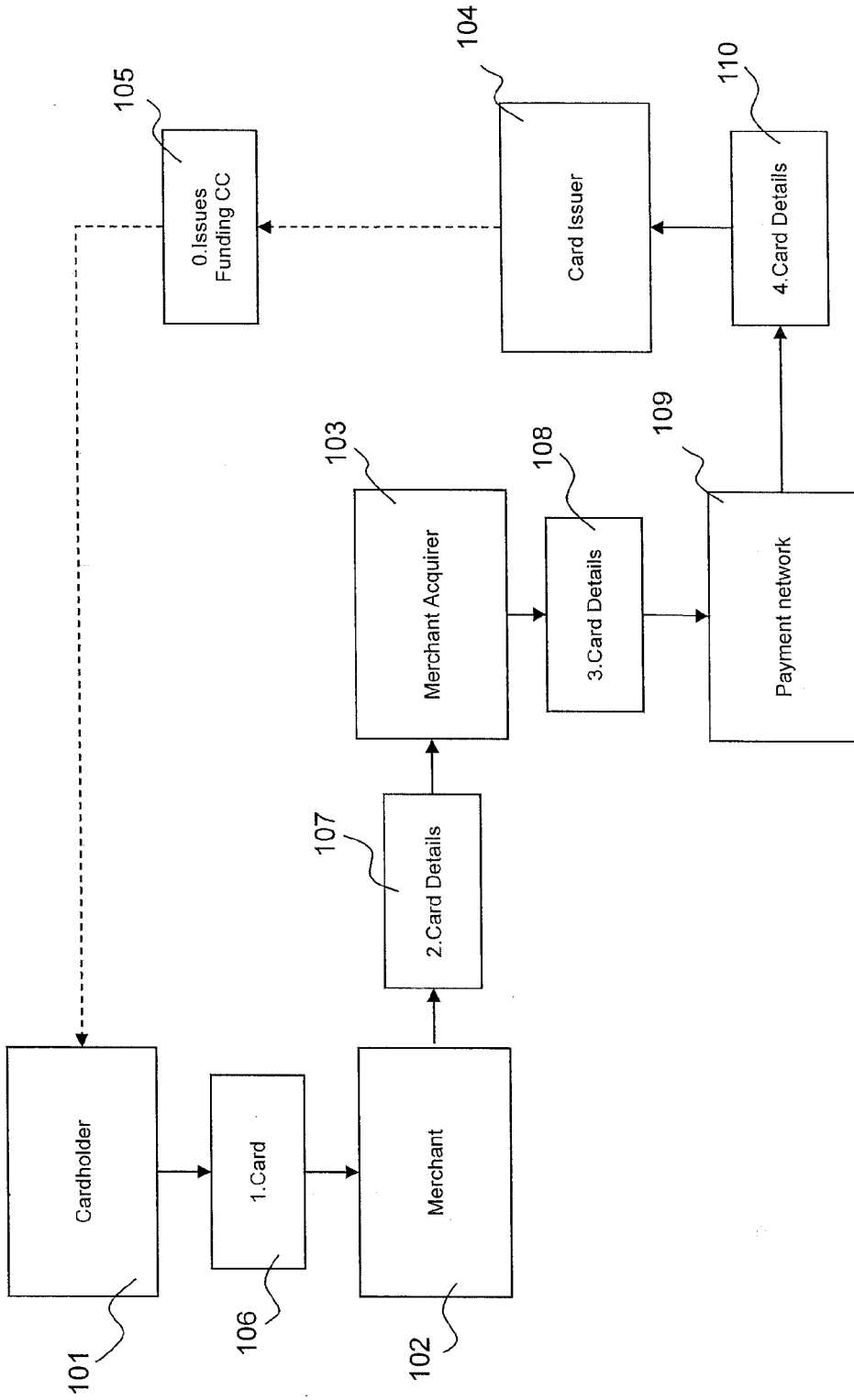


FIG. 1 (Prior Art)



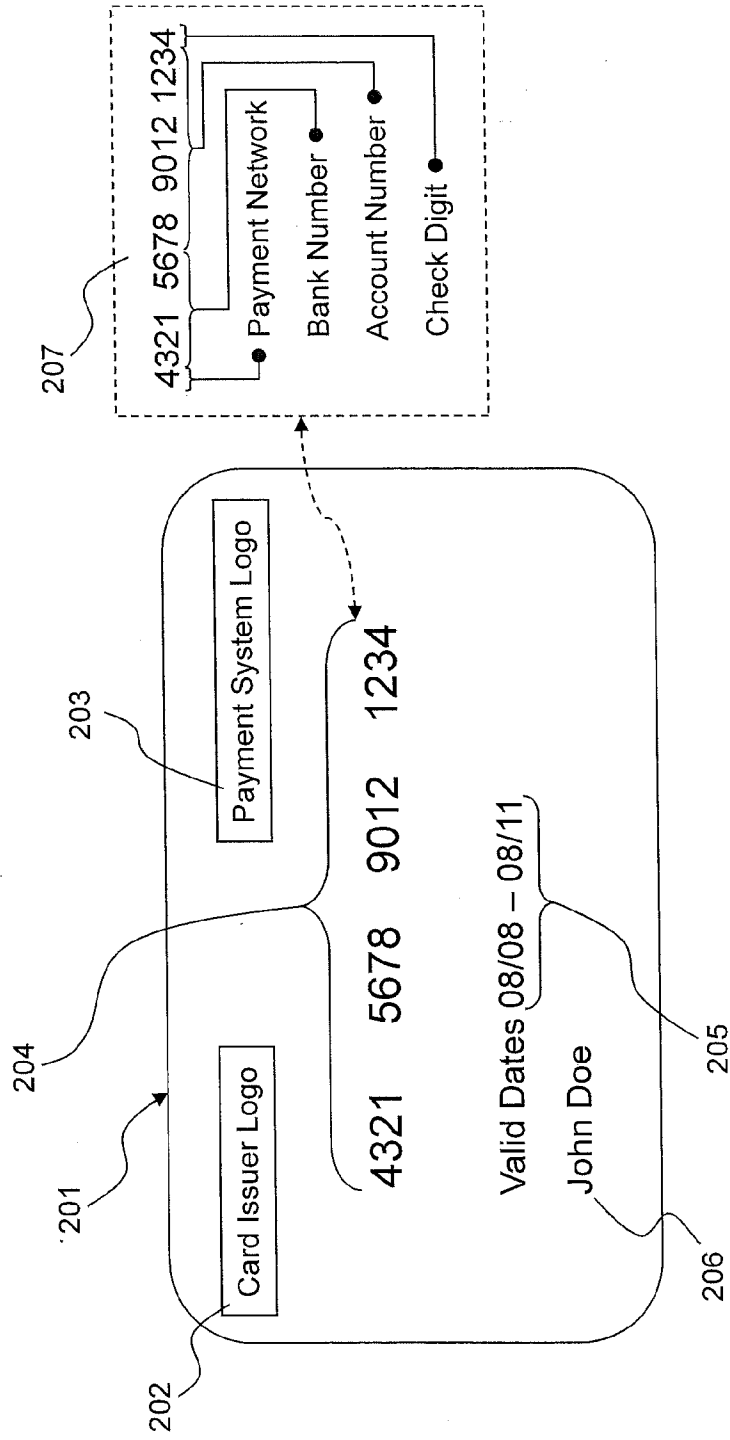


FIG. 2 (Prior Art)

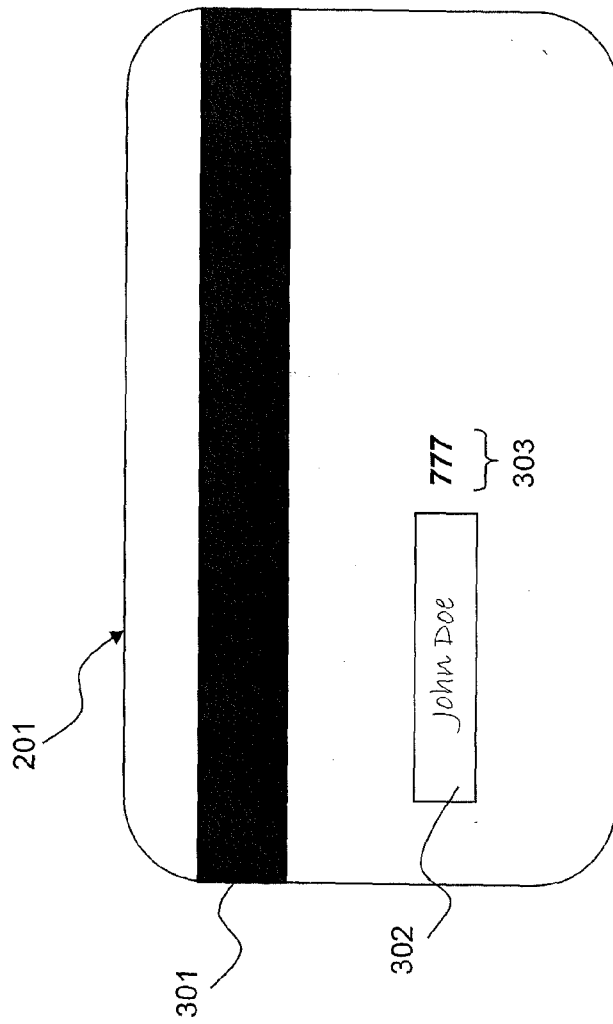


FIG. 3 (Prior Art)

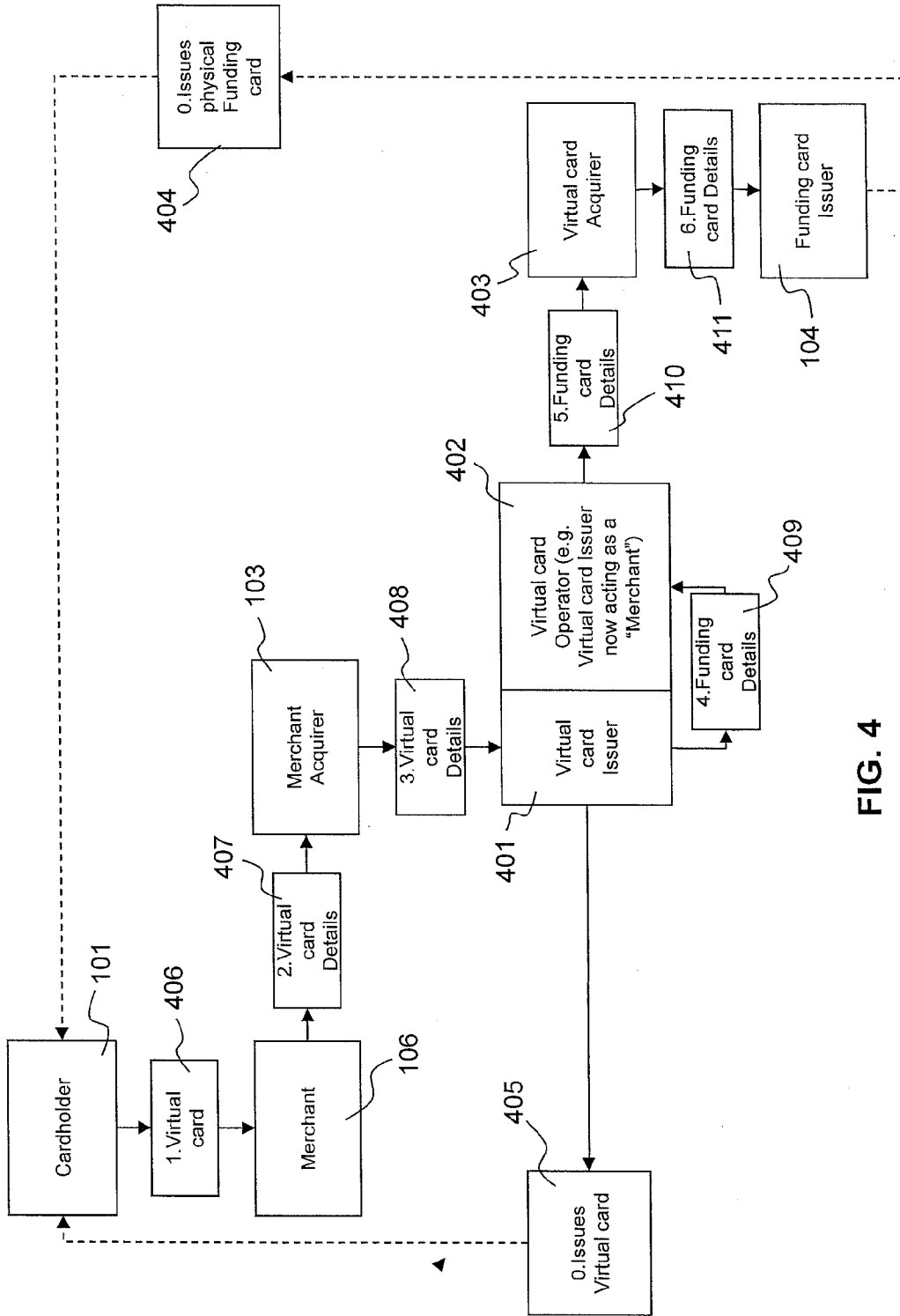


FIG. 4

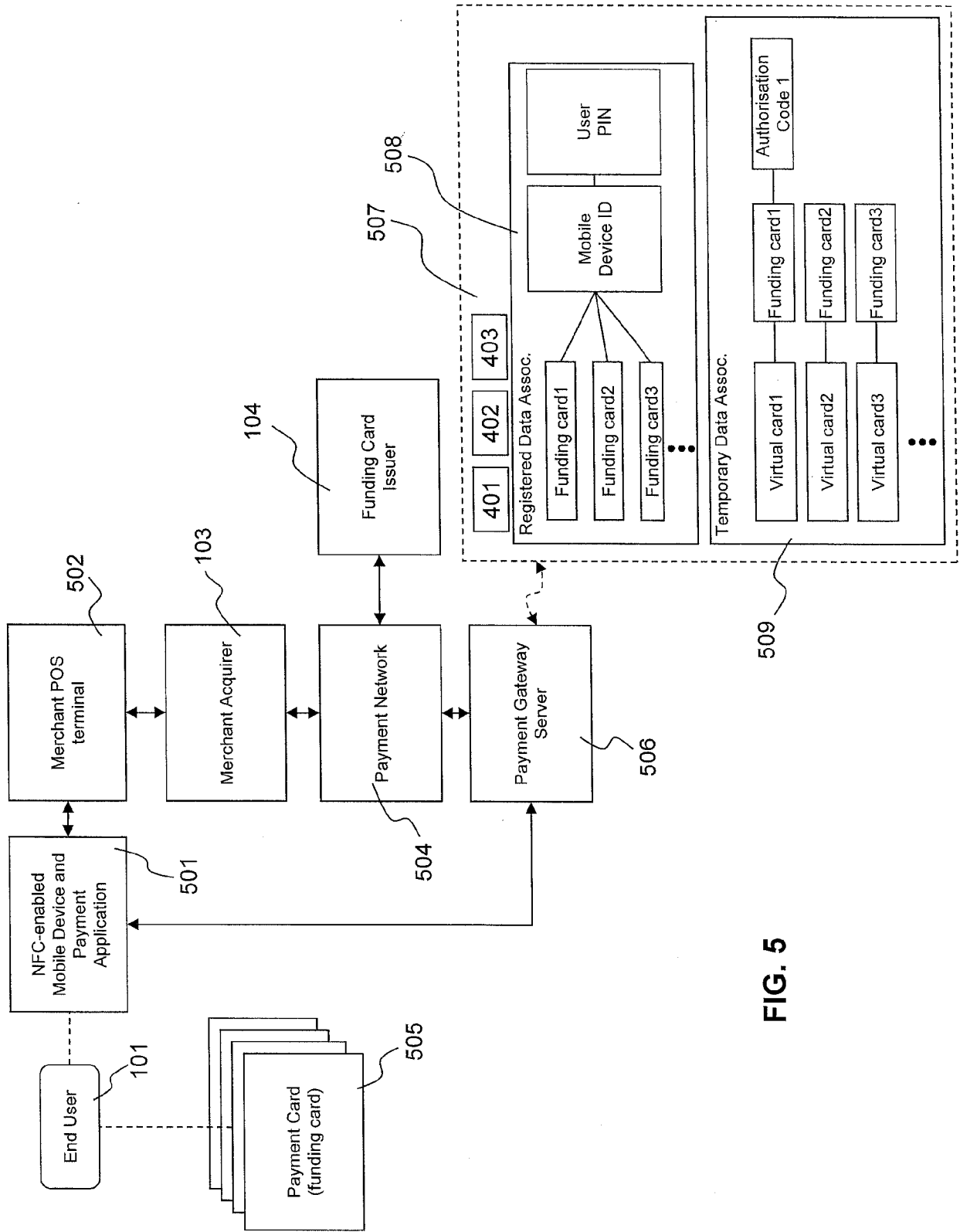


FIG. 5

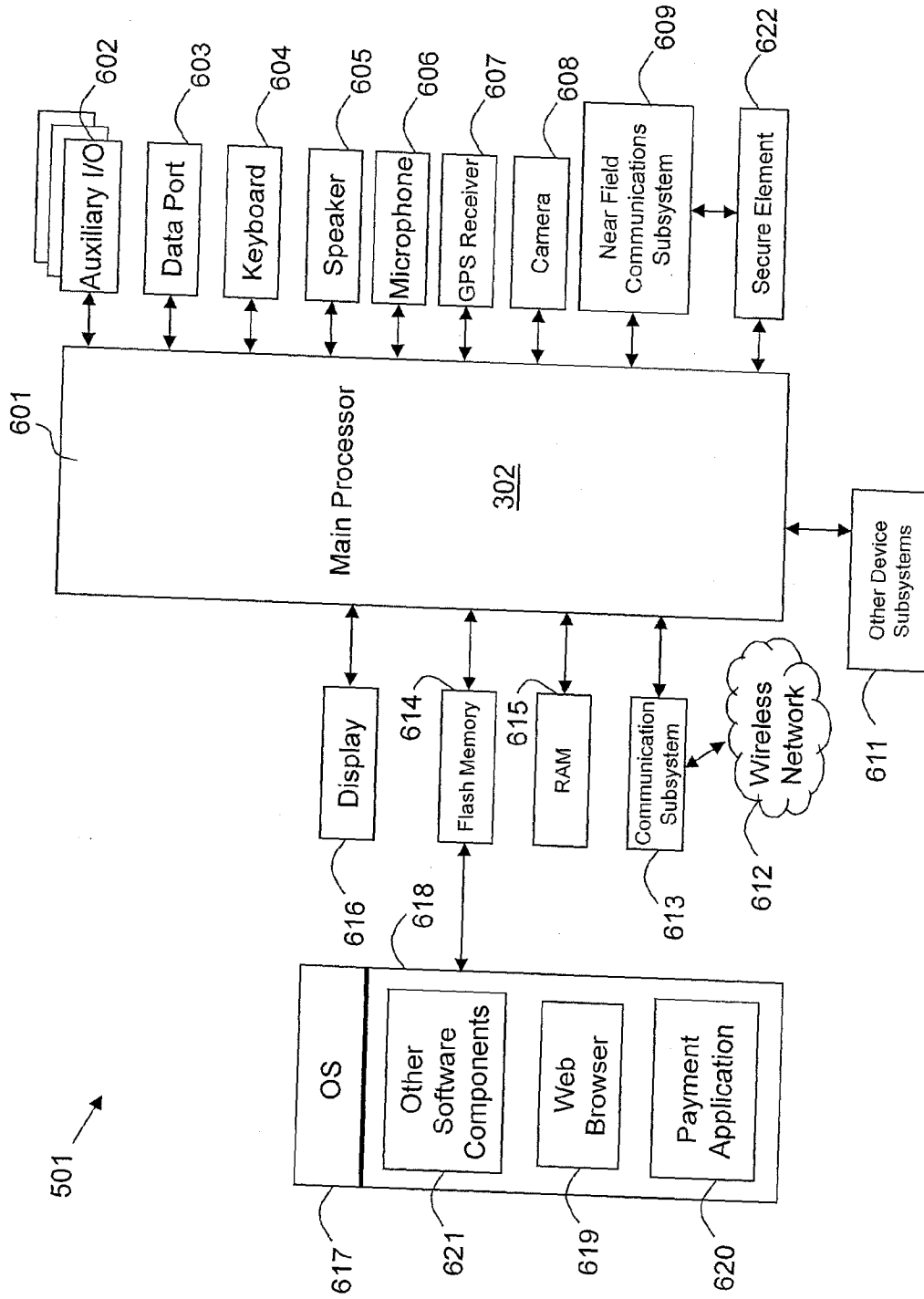


FIG. 6

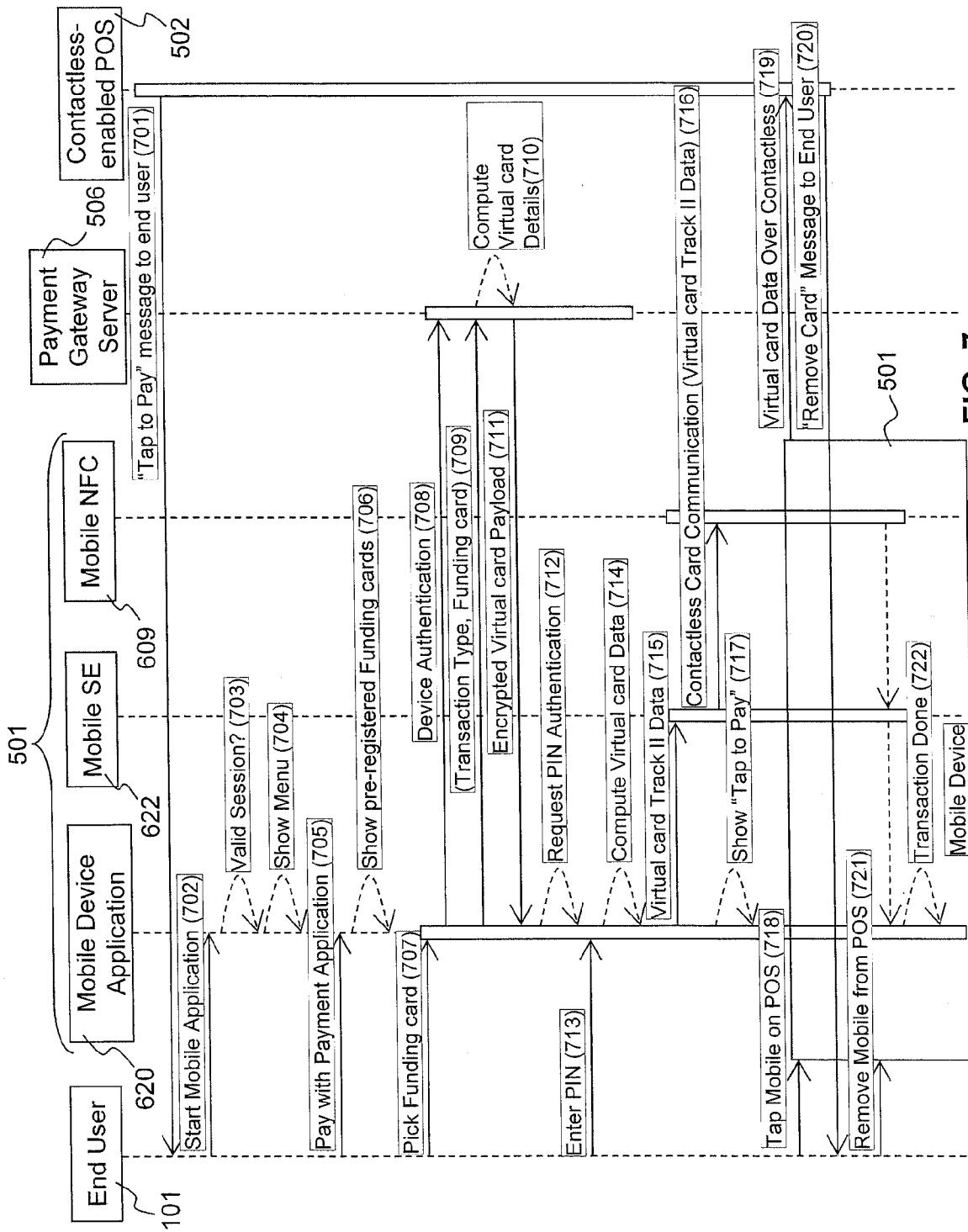


FIG. 7

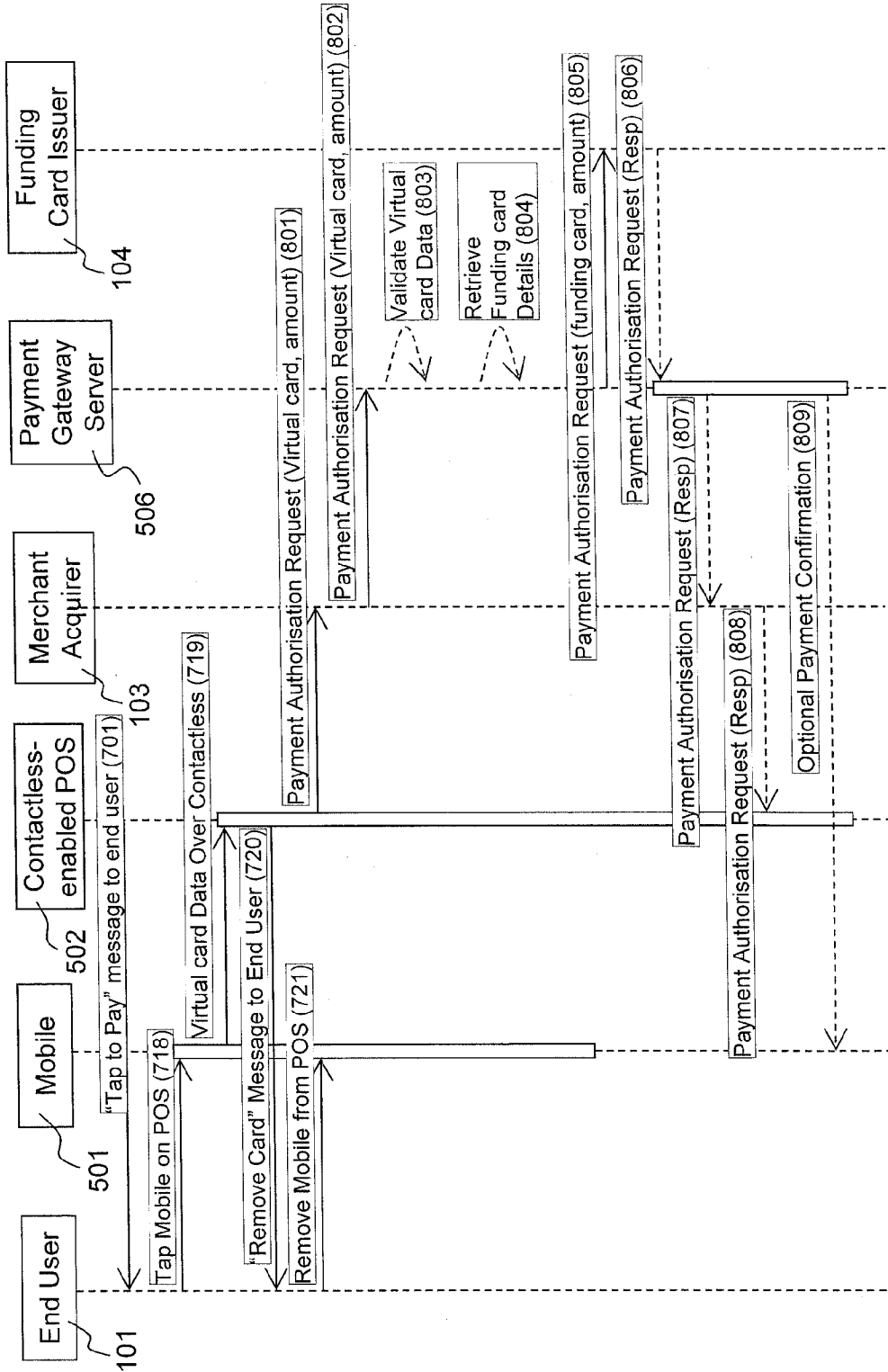


FIG. 8

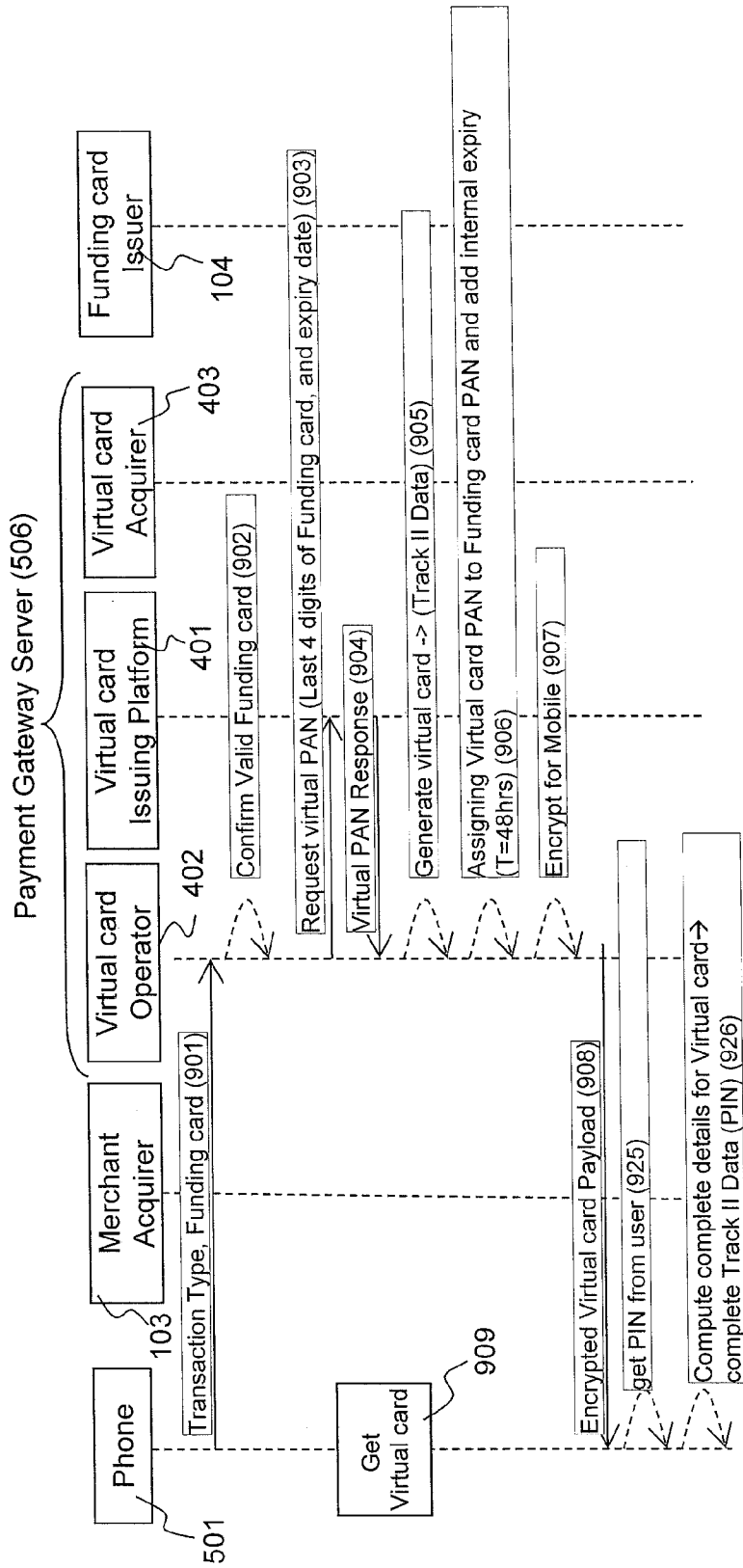


FIG. 9a



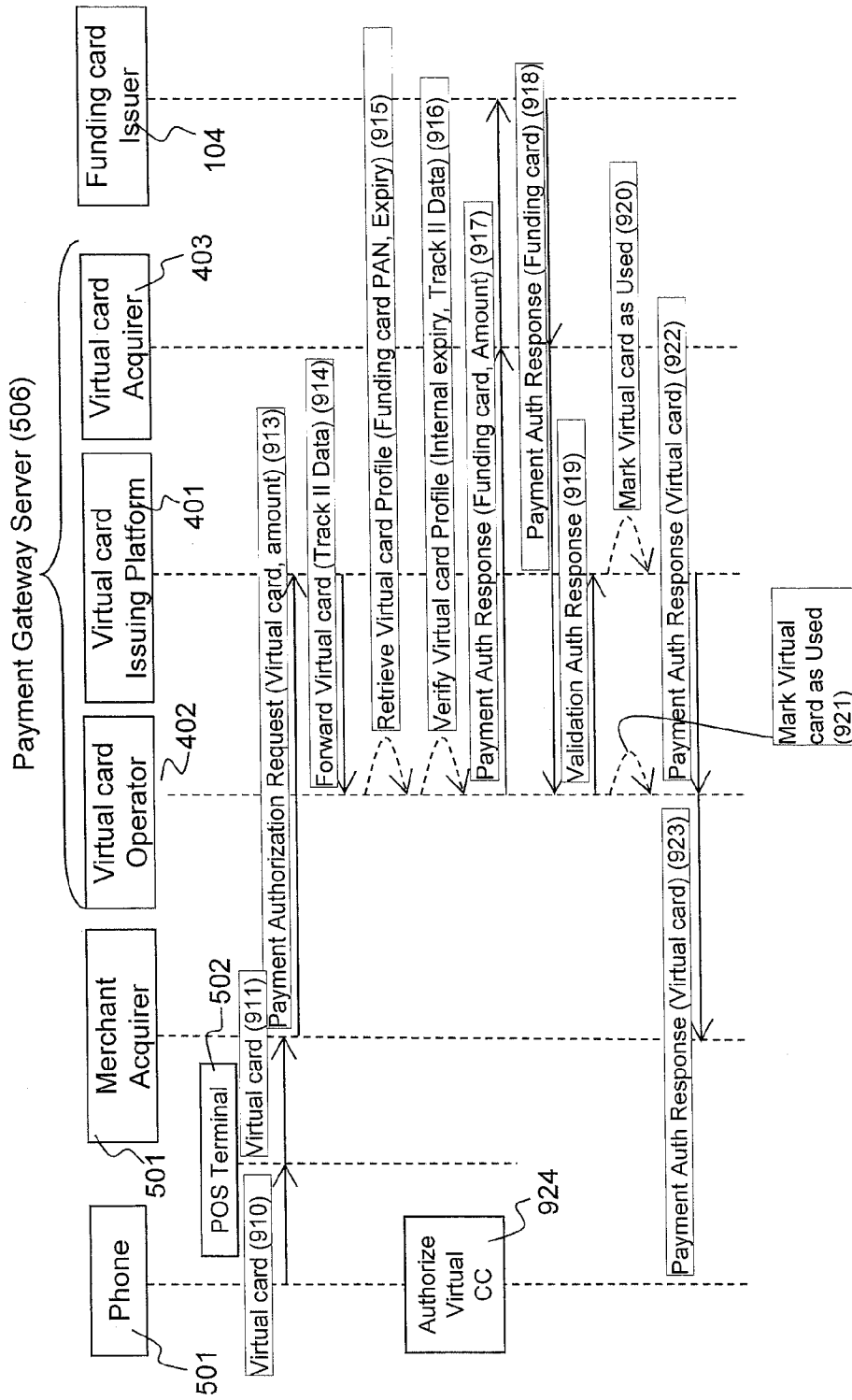


FIG. 9b

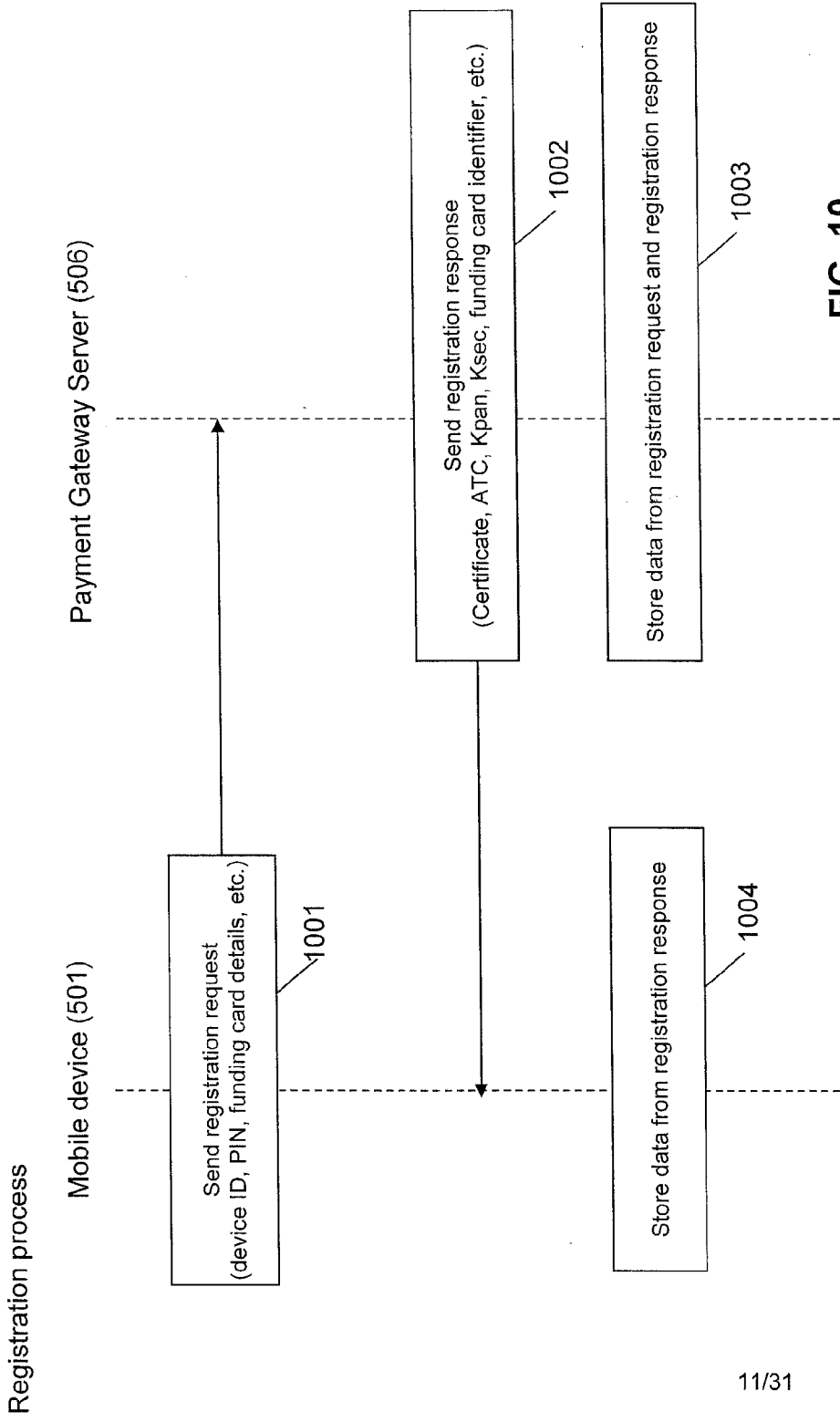


FIG. 10

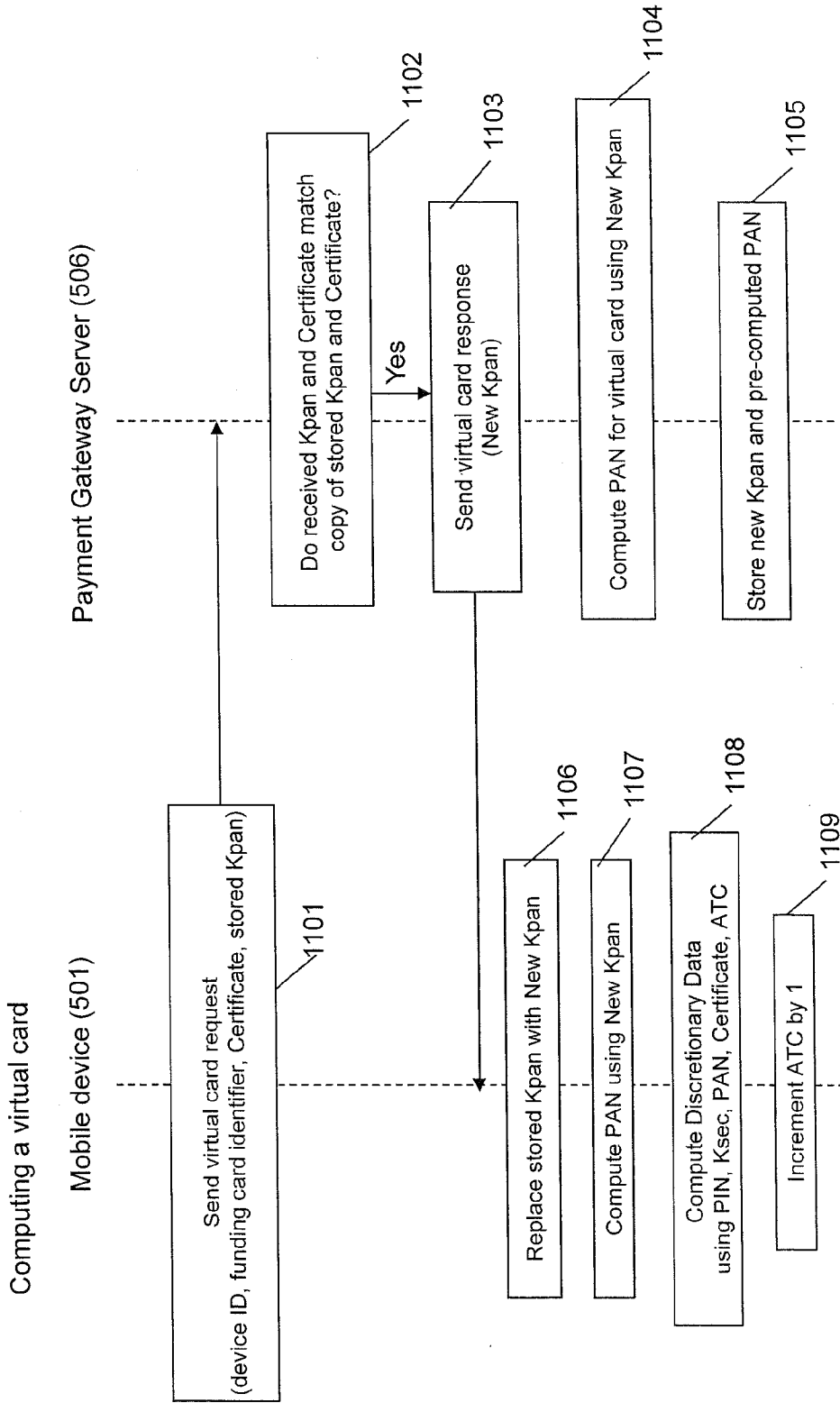


FIG. 11

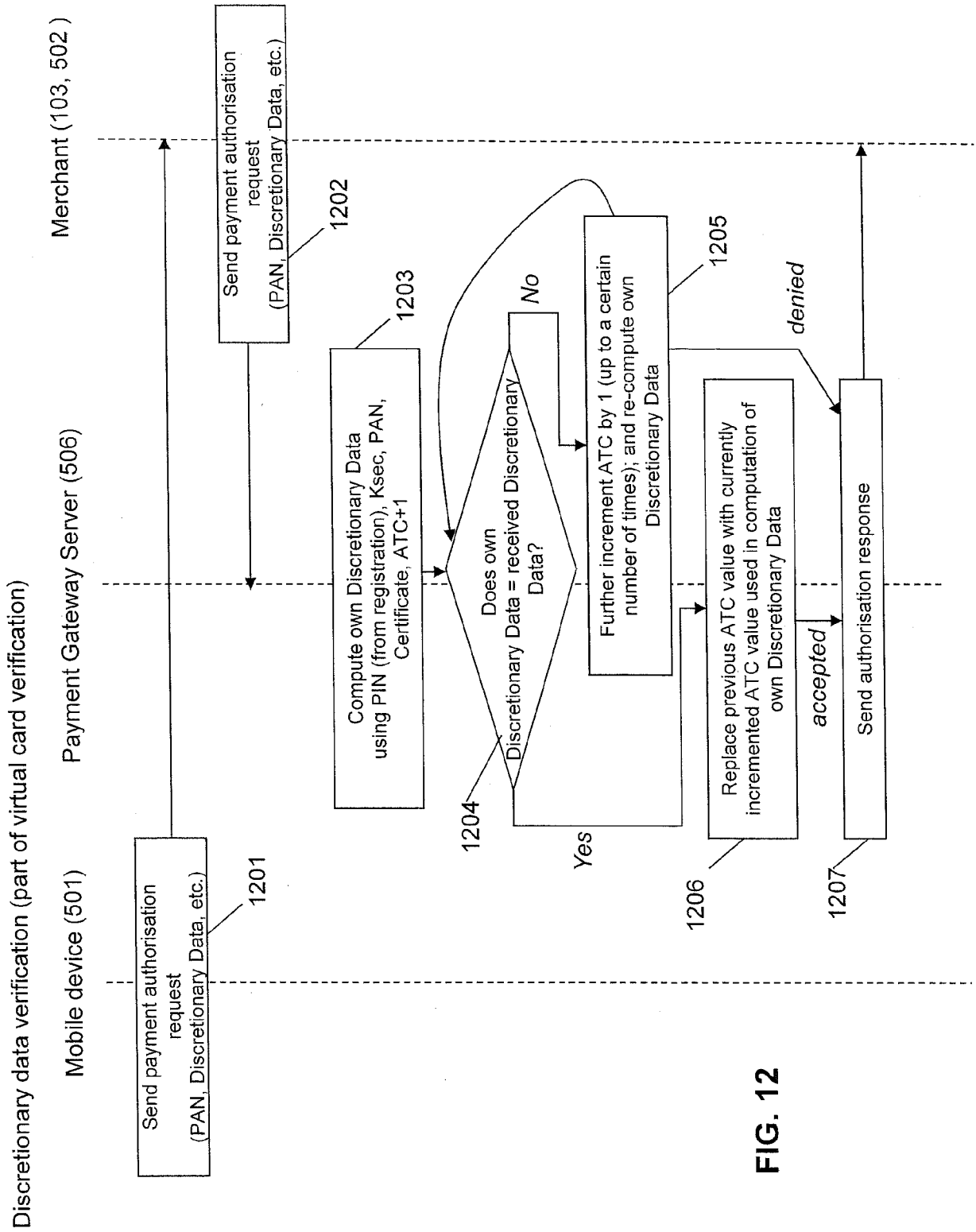


FIG. 12

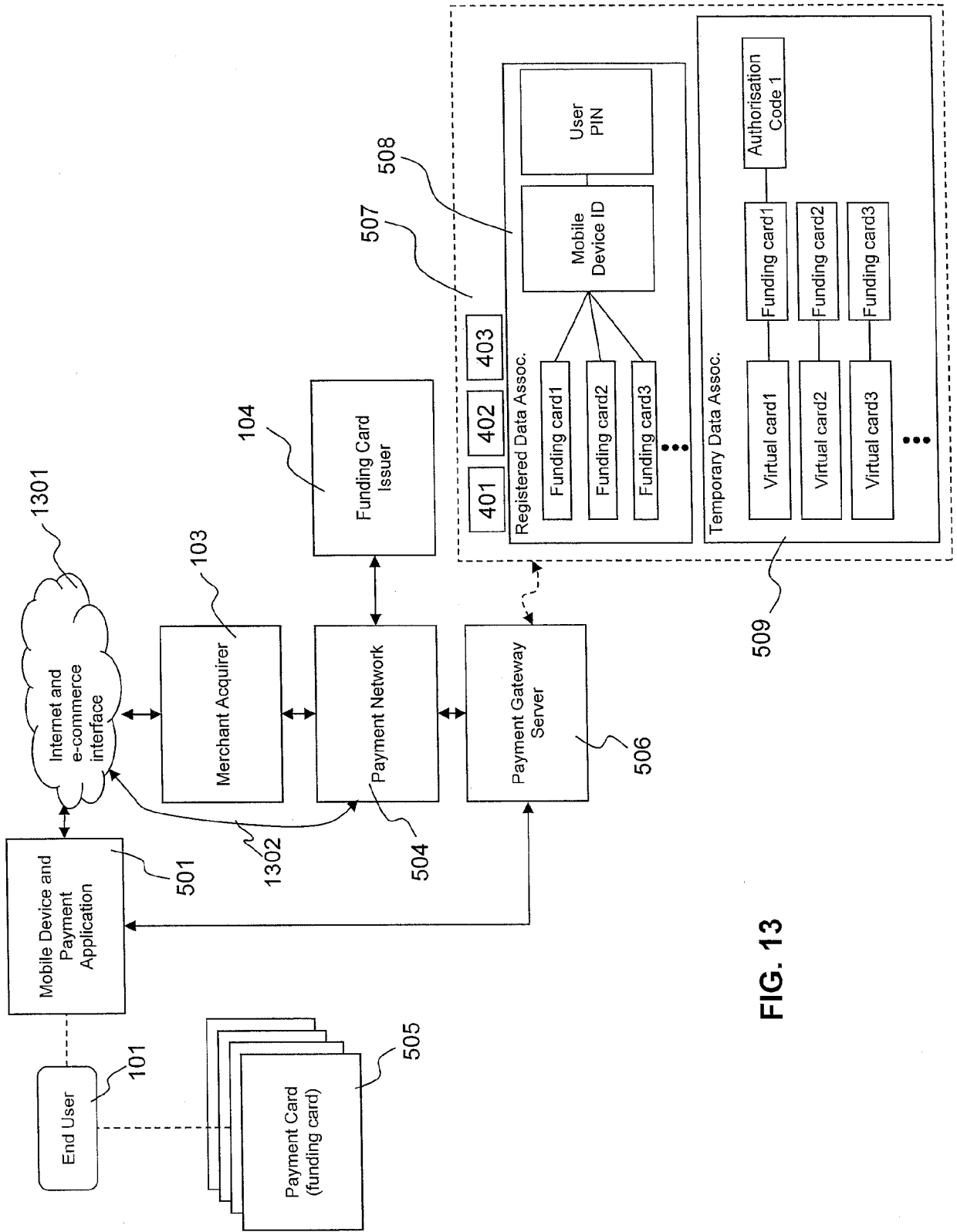



FIG. 13

# Merchant Store

**Payment** | Details | Address


**\$139.00** 1402

**John Doe**  
1291133566904

 **Paying with saved Visa** 1403  
Last 4 digits: 4242 1404

**Enter your PIN to complete purchase:**

**PIN:**  1405

 **Pay Now With OneTouch** 1406

OneTouch lets you make purchases without reentering your credit card each time.

[Change my credit card...](#) 1407




FIG. 14

Example GUI

1401 →

Example GUI

1501 →

16/31

**Merchant Store**

**Payment** | **Details** | **Address**

**\$139.00** 1402




**John Doe**  
1291133566904

**VISA** **Paying with saved Visa** 1403  
Last 4 digits: 4242 1404

**Pay Now With One Touch** 1406

OneTouch lets you make purchases without reentering your credit card each time.

Change my credit card... 1407

**FIG. 15**

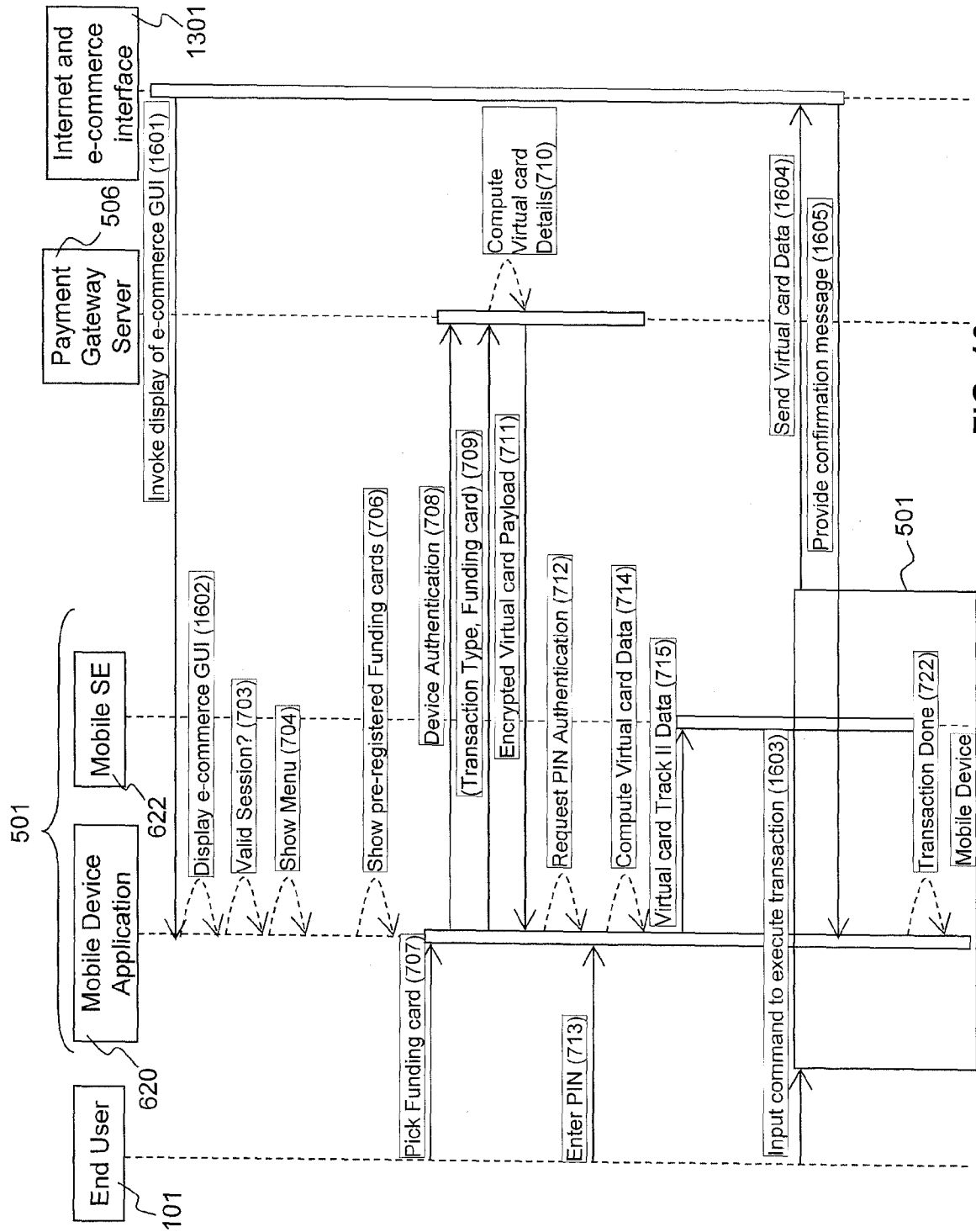


FIG. 16



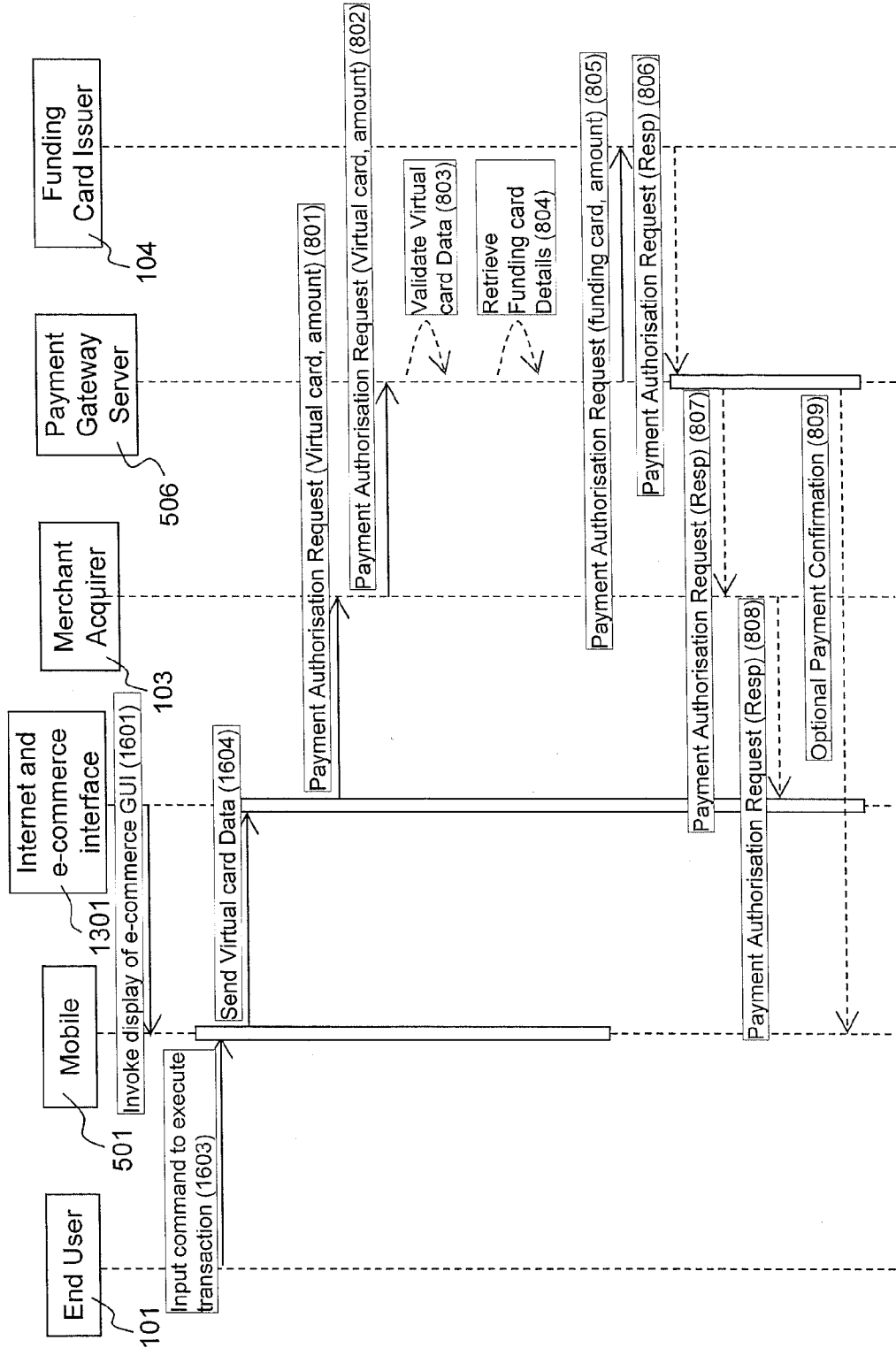


FIG. 17

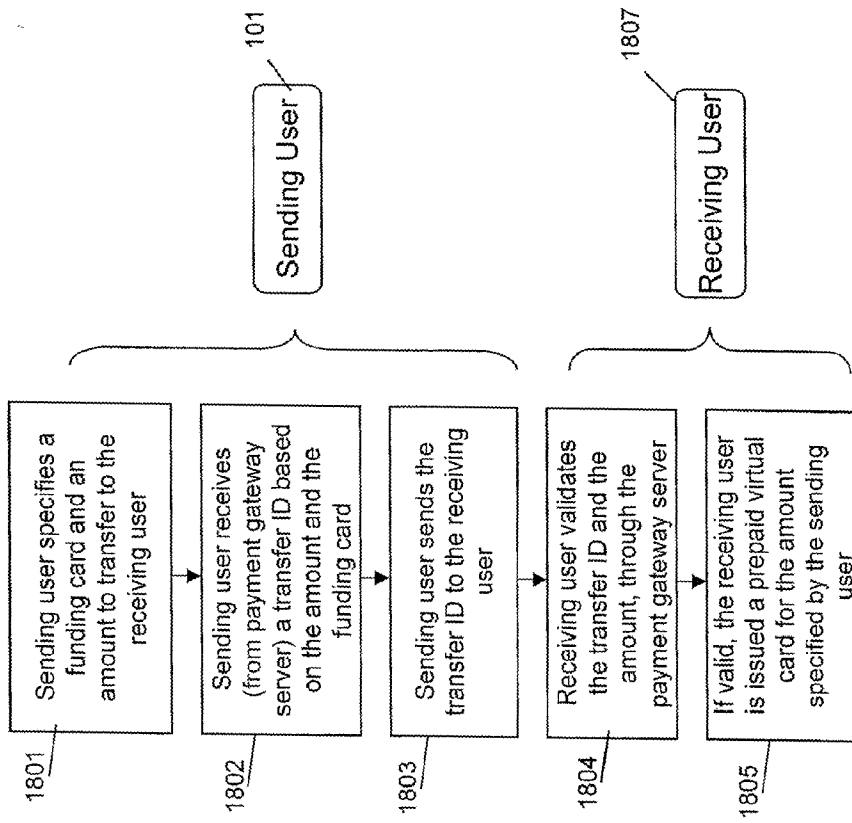


FIG. 18

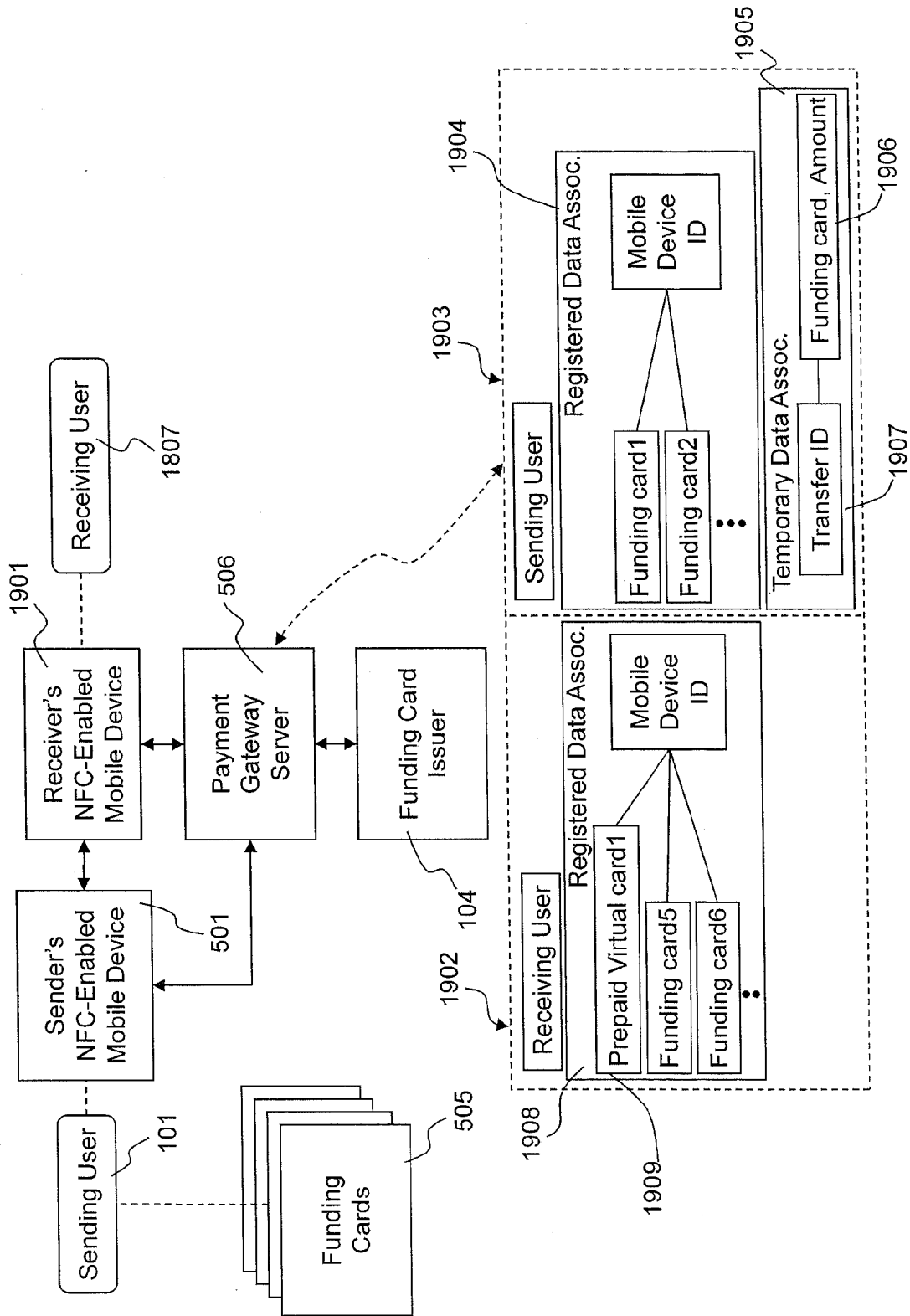


FIG. 19

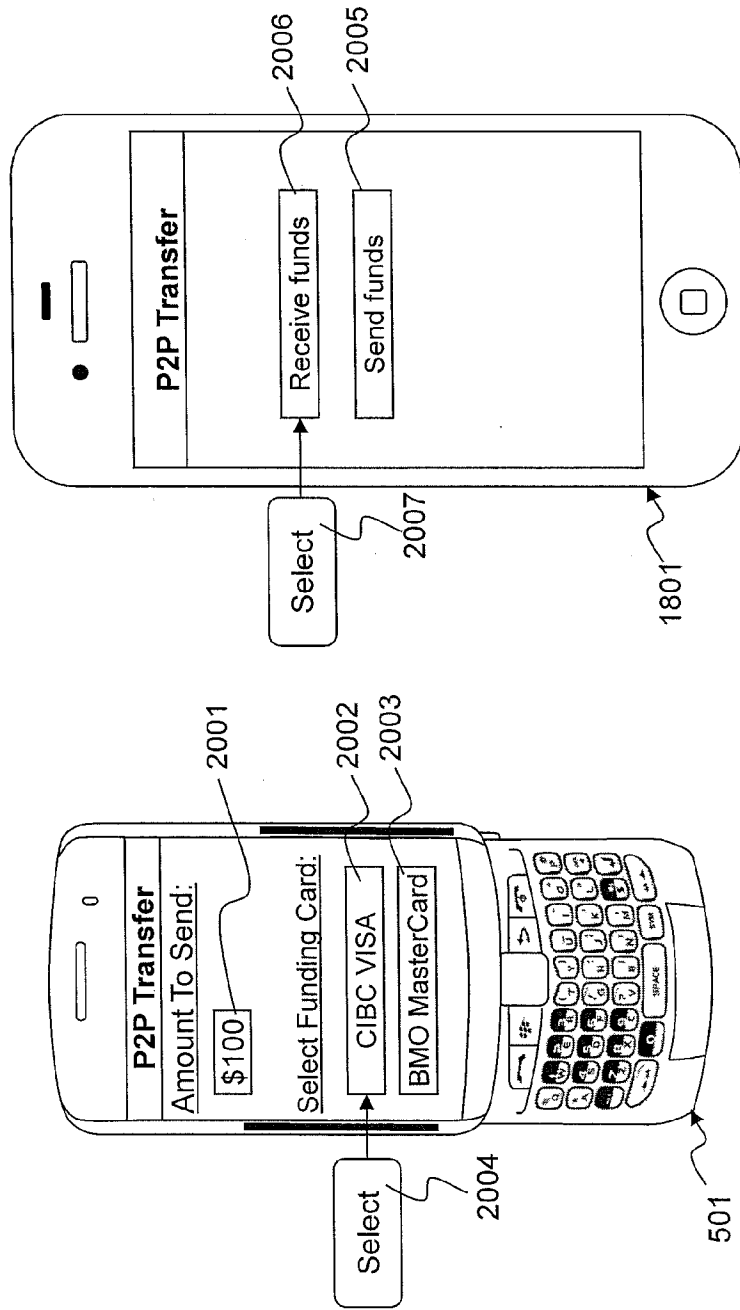


FIG. 20

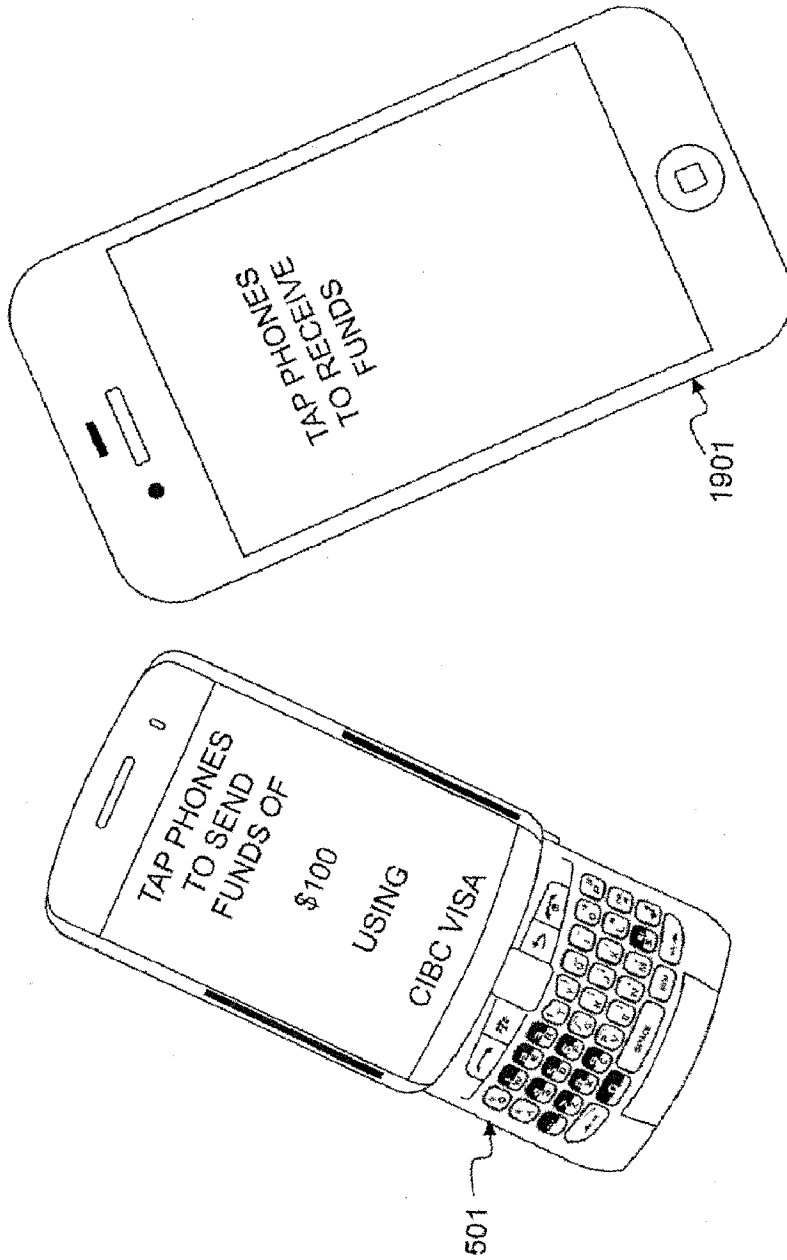


FIG. 21

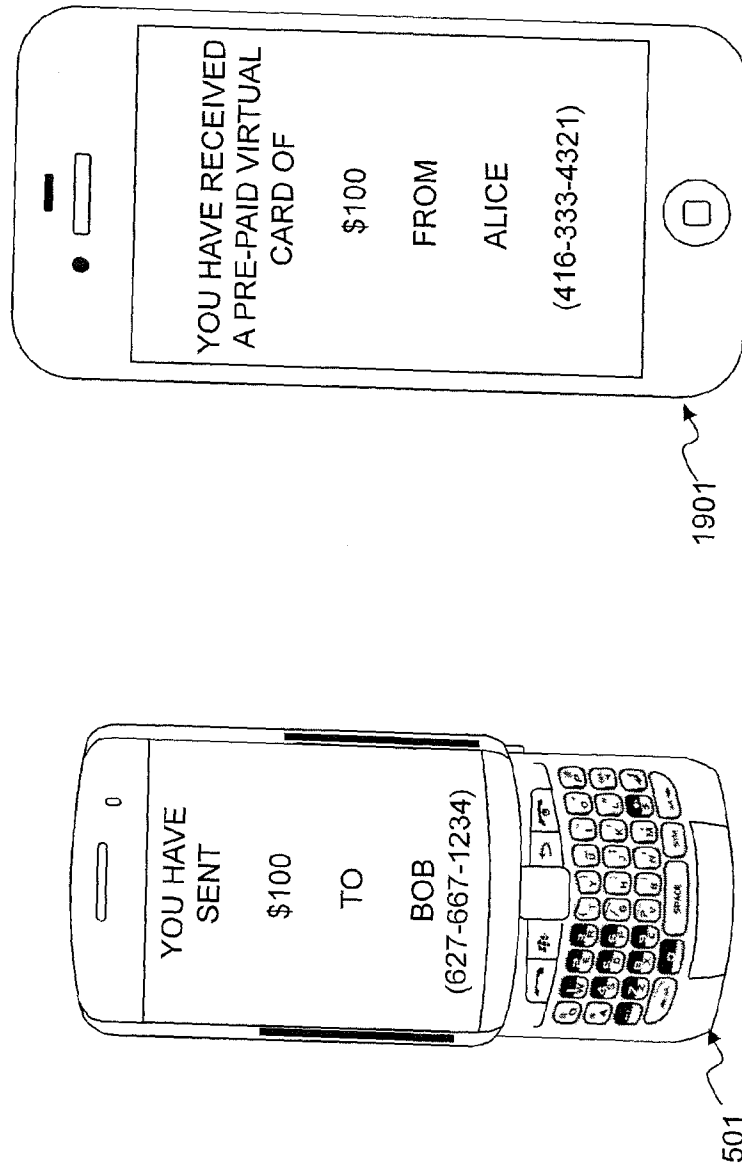


FIG. 22

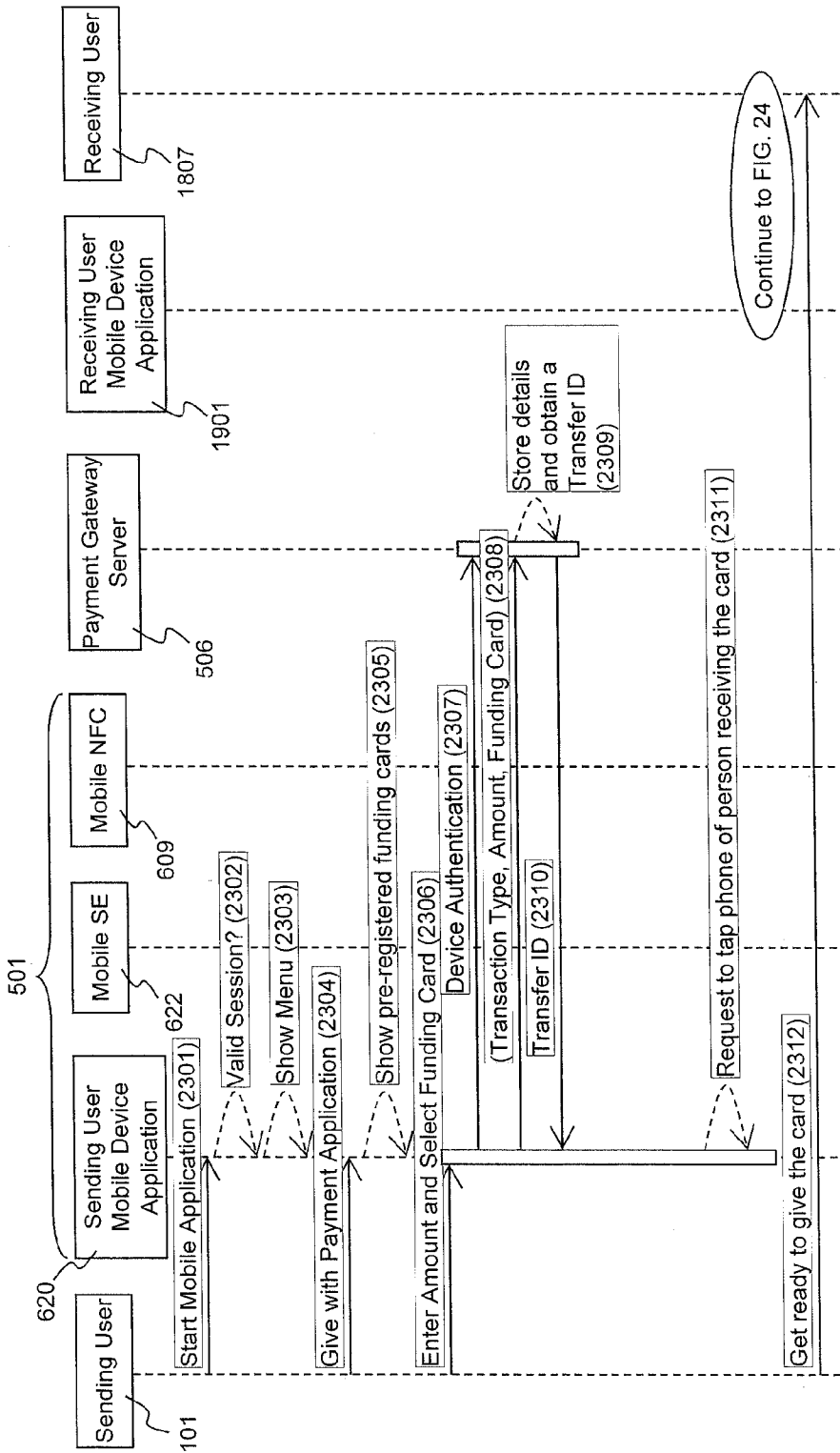


FIG. 23

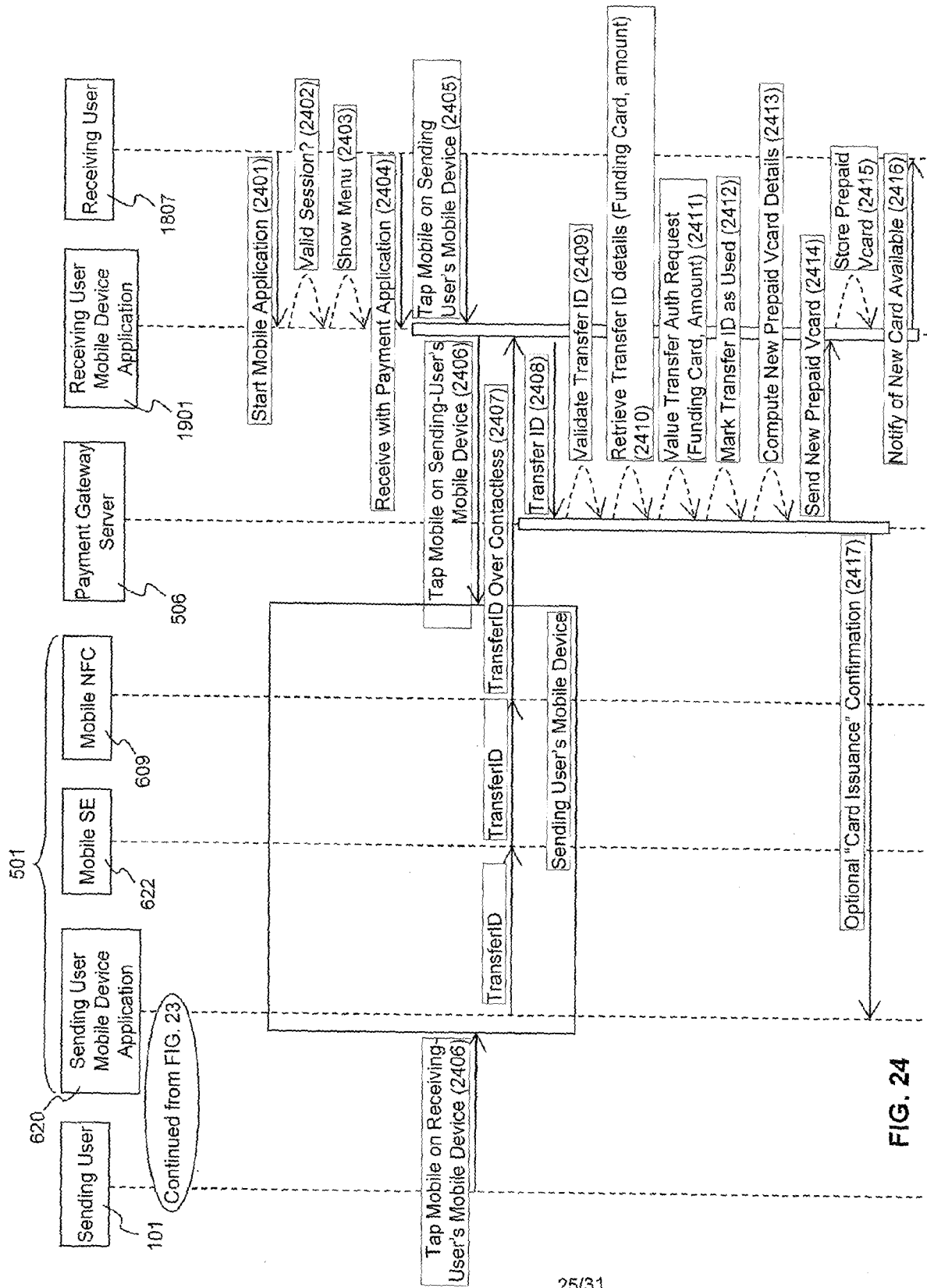


FIG. 24



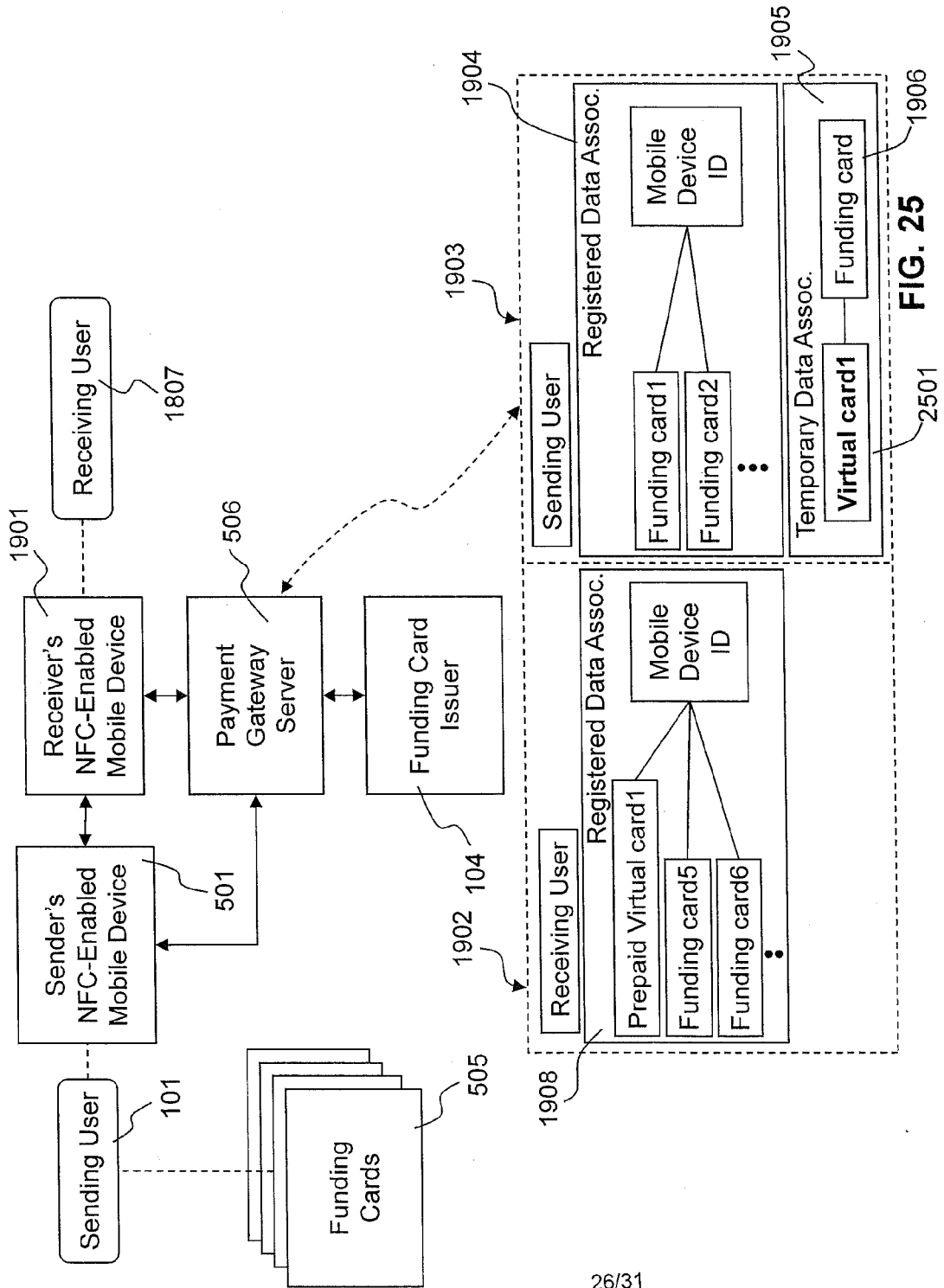


FIG. 25

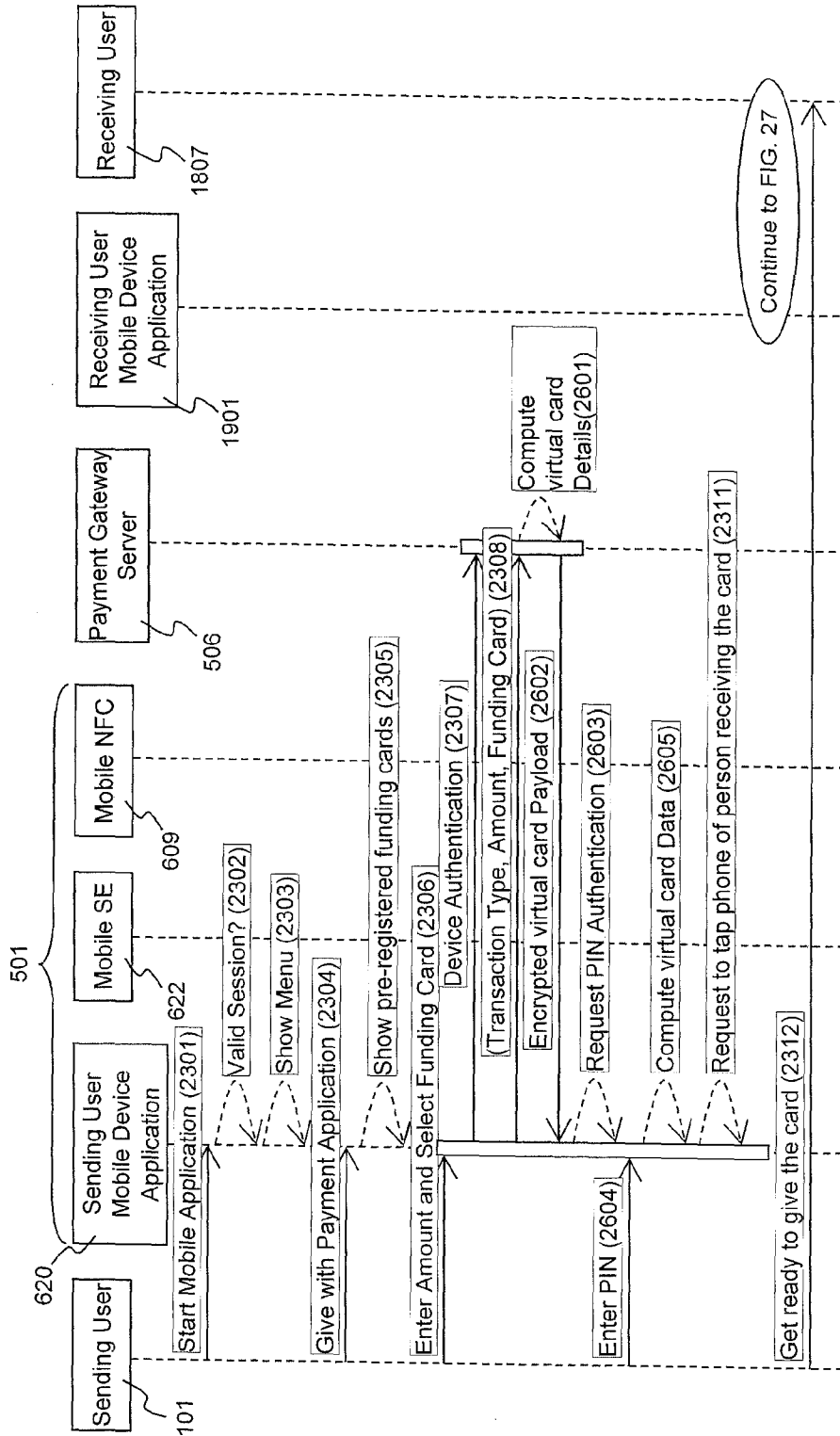


FIG. 26

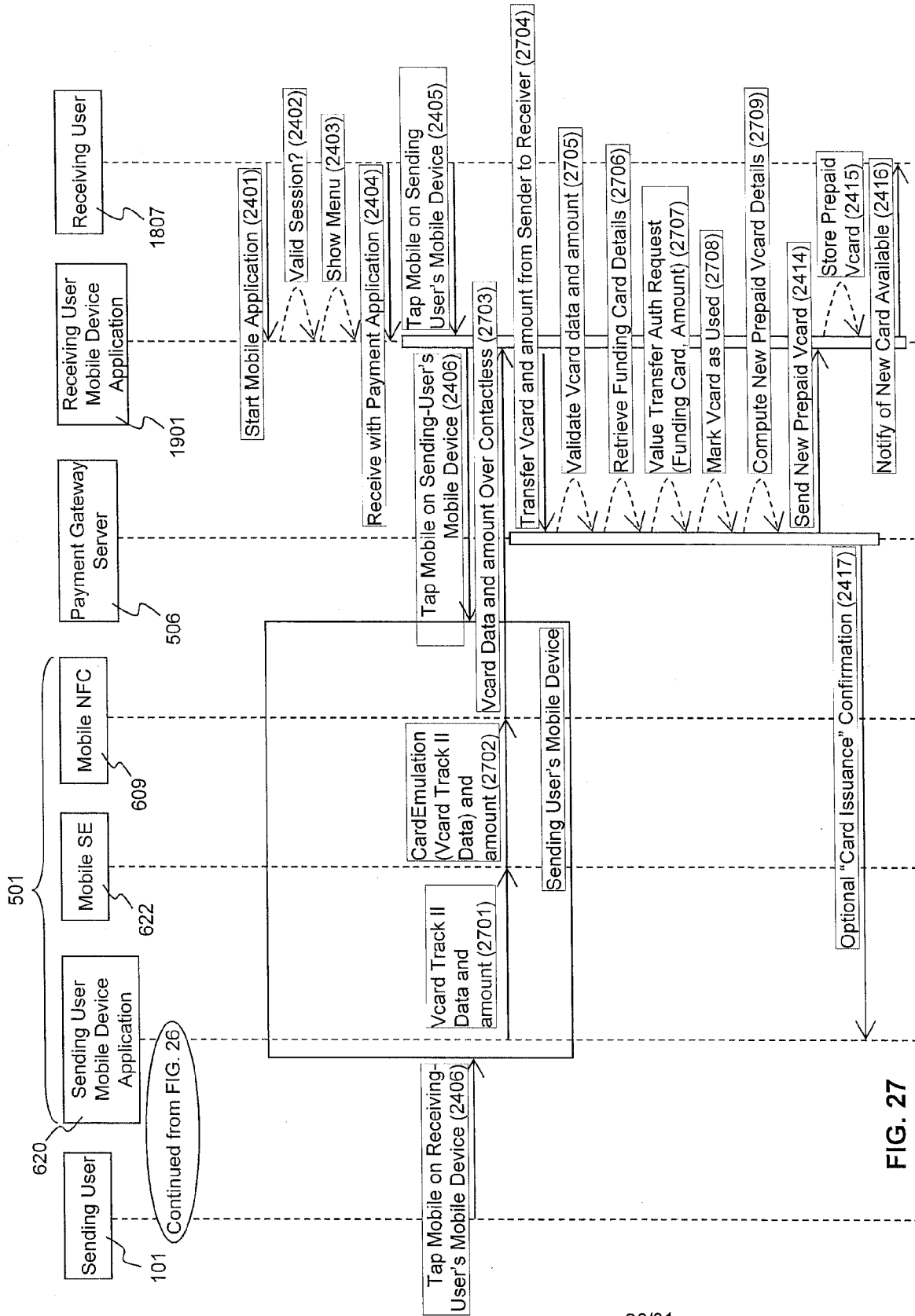


FIG. 27

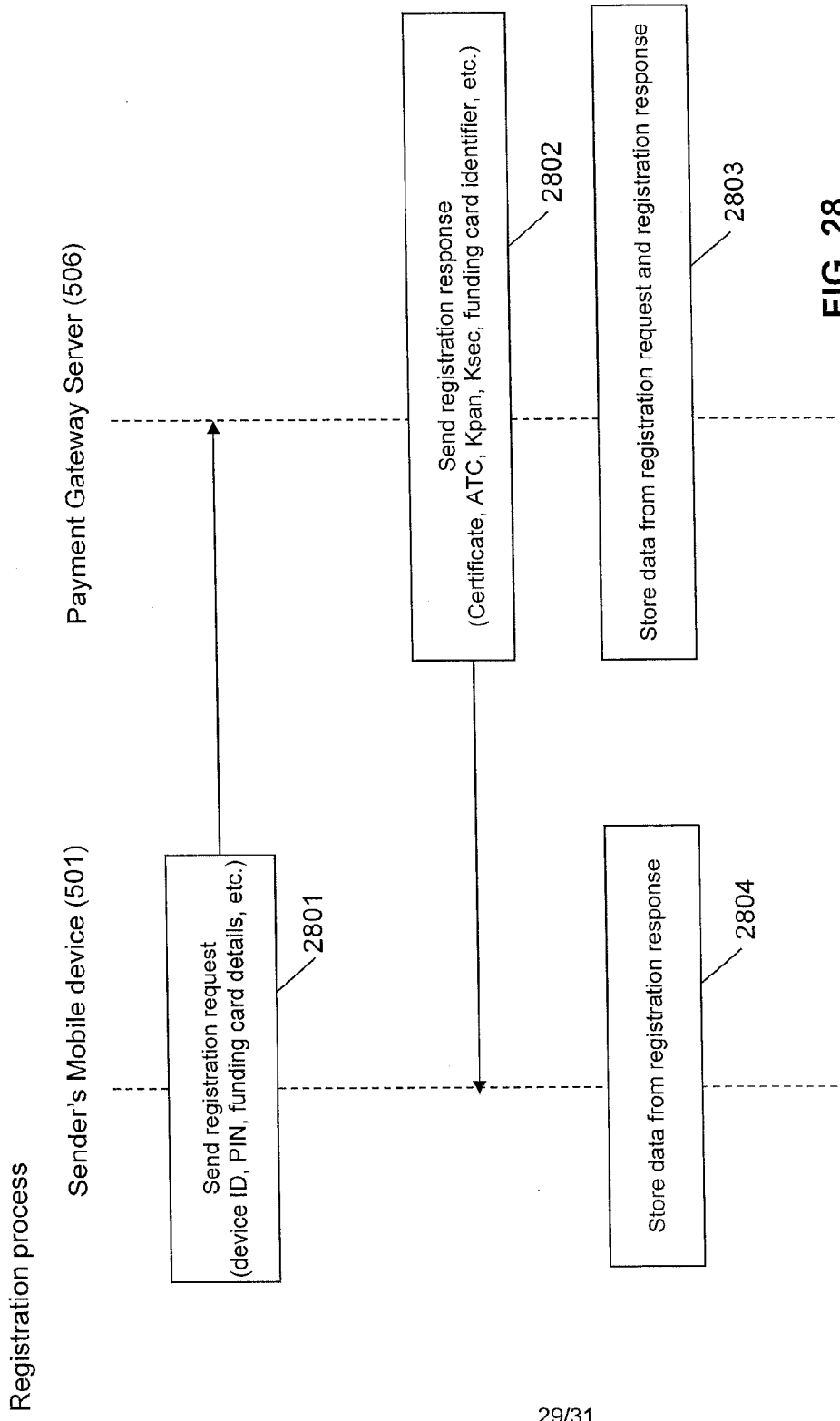


FIG. 28

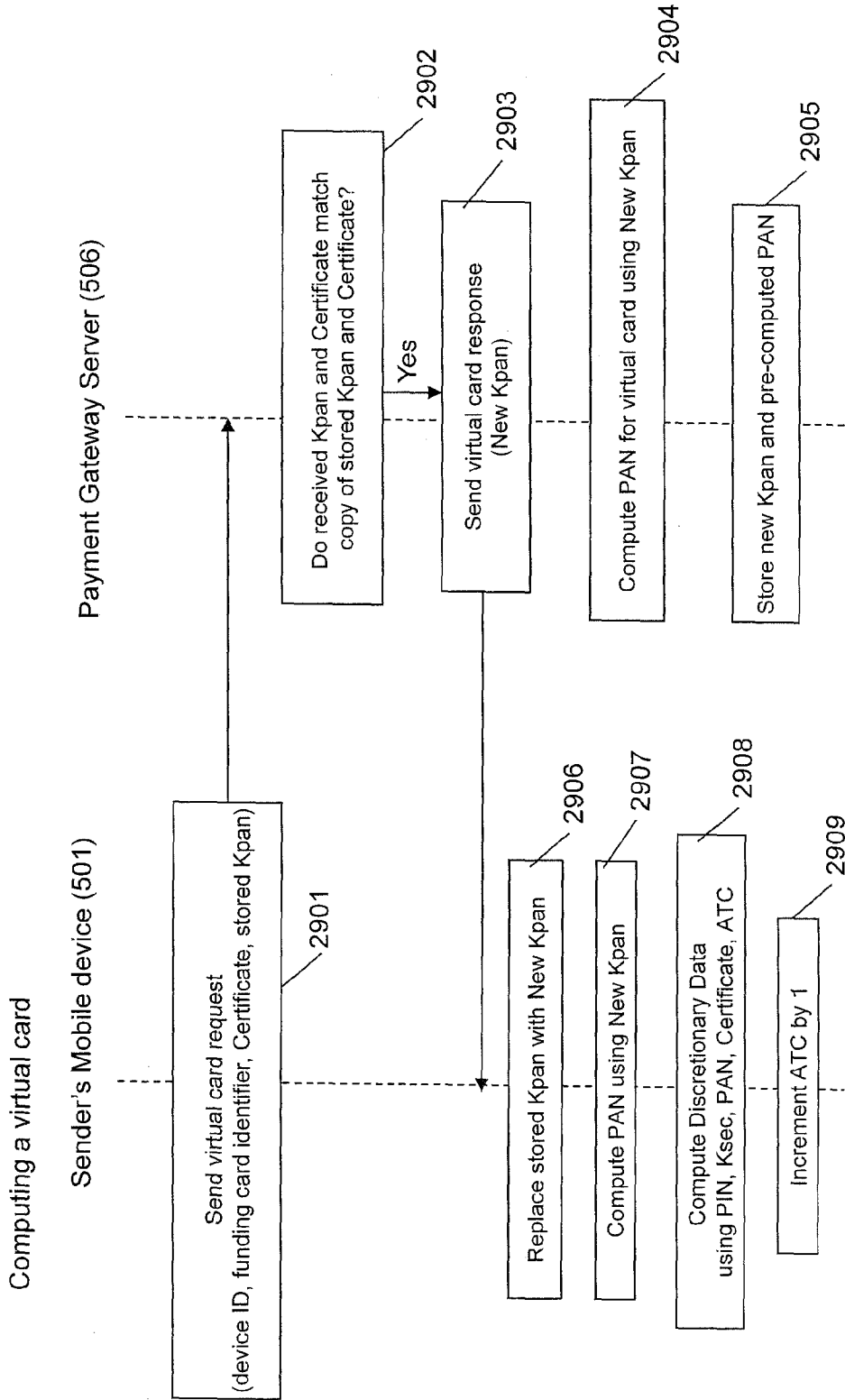


FIG. 29

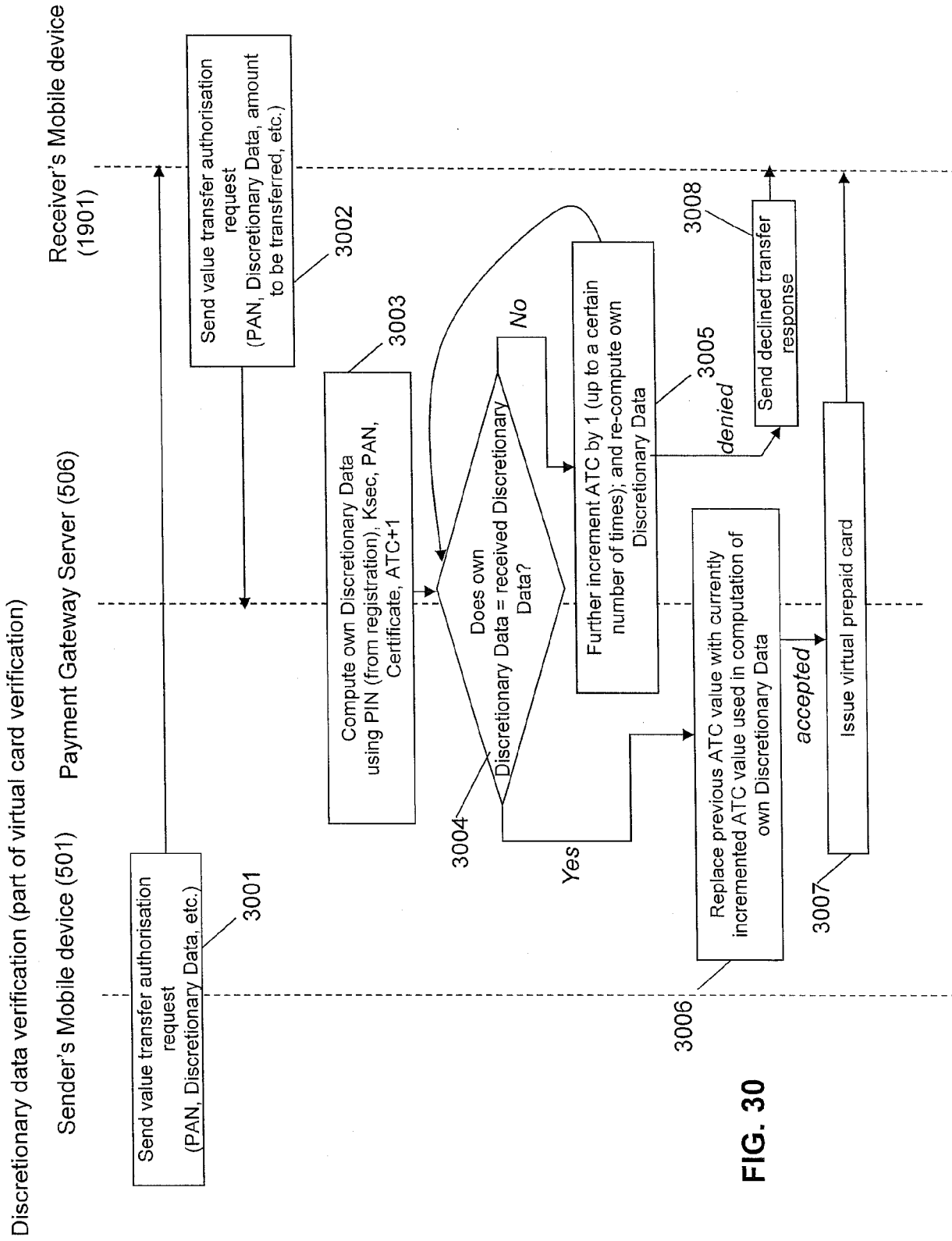


FIG. 30

**INTERNATIONAL SEARCH REPORT**

International application No.  
PCT/CA2013/050294

<p>A. CLASSIFICATION OF SUBJECT MATTER  <b>IPC: G06Q 20/34 (2012.01) , G06Q 20/32 (2012.01) , G06Q 20/40 (2012.01)</b>                  According to International Patent Classification (IPC) or to both national classification and IPC</p>											
<p>B. FIELDS SEARCHED</p> <p>Minimum documentation searched (classification system followed by classification symbols)  <b>IPC: G06Q</b></p> <p>Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched</p> <p>Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used)                  Total Patent, Google                  virtual card payment NFC gateway processor authorization security barcode credit</p>											
<p>C. DOCUMENTS CONSIDERED TO BE RELEVANT</p> <table border="1" style="width:100%; border-collapse: collapse;"> <thead> <tr> <th style="width:10%;">Category*</th> <th style="width:60%;">Citation of document, with indication, where appropriate, of the relevant passages</th> <th style="width:30%;">Relevant to claim No.</th> </tr> </thead> <tbody> <tr> <td align="center">X</td> <td>WO 03/023674 A1 (UM) 20 April 2003 (20-04-2003) *figs. 1, 3 - 4*</td> <td align="center">1 - 3</td> </tr> <tr> <td align="center">X</td> <td>WO 2012/014231 A1 (JUTHANI) 02 February 2012 (02-02-2012) *figs. 19 - 20*</td> <td align="center">1 - 3</td> </tr> </tbody> </table>			Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.	X	WO 03/023674 A1 (UM) 20 April 2003 (20-04-2003) *figs. 1, 3 - 4*	1 - 3	X	WO 2012/014231 A1 (JUTHANI) 02 February 2012 (02-02-2012) *figs. 19 - 20*	1 - 3
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.									
X	WO 03/023674 A1 (UM) 20 April 2003 (20-04-2003) *figs. 1, 3 - 4*	1 - 3									
X	WO 2012/014231 A1 (JUTHANI) 02 February 2012 (02-02-2012) *figs. 19 - 20*	1 - 3									
<p><input type="checkbox"/> Further documents are listed in the continuation of Box C.</p>		<p><input checked="" type="checkbox"/> See patent family annex.</p>									
<p>* Special categories of cited documents :</p> <p>"A" document defining the general state of the art which is not considered to be of particular relevance</p> <p>"E" earlier application or patent but published on or after the international filing date</p> <p>"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)</p> <p>"O" document referring to an oral disclosure, use, exhibition or other means</p> <p>"P" document published prior to the international filing date but later than the priority date claimed</p>	<p>"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention</p> <p>"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone</p> <p>"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art</p> <p>"&amp;" document member of the same patent family</p>										
<p>Date of the actual completion of the international search                  19 July 2013 (19-07-2013)</p>		<p>Date of mailing of the international search report                  12 August 2013 (12-08-2013)</p>									
<p>Name and mailing address of the ISA/CA                  Canadian Intellectual Property Office                  Place du Portage I, C114 - 1st Floor, Box PCT                  50 Victoria Street                  Gatineau, Quebec K1A 0C9                  Facsimile No.: 001-819-953-2476</p>		<p>Authorized officer  <b>Minghui Shi</b>                  819-997-7623</p>									

**INTERNATIONAL SEARCH REPORT**  
Information on patent family members

International application No.  
**PCT/CA2013/050294**

Patent Document Cited in Search Report	Publication Date	Patent Family Member(s)	Publication Date
WO03023674A1	20-03-2003	KR20010090081A	18-10-2001
WO2012014231A1	02-02-2012	EP2598984A1	05-06-2013
		US2013124292A1	16-05-2013
		WO2012014231A4	07-06-2012