eGiftSolutions Message Format API – Version G1

eGiftSolutions is a gift/loyalty platform currently supporting more than 50+ integrations and also accepting processing online via virtual terminal and mobile app. In addition to the direct certifications, it supports Datacap GiftEPay for processing in numerous POS systems. This guide will give you the needed information to submit packets into eGiftSolutions for processing and will allow you to integrate your application into the eGiftSolutions platform. For support, you can work with your Integration Support Manager or email support@egiftsolutions.com for assistance.

Table of Contents:

1. General Considerations – Start Here

Common

- When you are ready to start submitting test transactions into eGiftSolutions to complete your certification process, we will assign your software/terminal a Terminal Type which will be submitted with all packets from your software/terminal. This allows reporting to be organized and identify your terminal and/or software submitting the transaction request. Submit an email to support@egiftsolutions.com for a Terminal Type request or work with your Integration Support Manager for assistance.

- **Amount 1** and **Amount 2 fields**, both in terminal requests and server responses, are ASCII numeric strings representing a currency amount expressed in cents *with no currency symbol, comma, or decimal point* – or a loyalty point amount expressed as an unsigned integer . However, the

server response often contains a display-formatted amount string in the **Text** field.

- Data travel in packets. For all transactions except history, a request (terminal to server) is one packet and a response (server to terminal) is one packet.

- Each packet begins with STX (hex 02) and ends with ETX (hex 03) followed by LRC (variable).

- Fields within each packet are delimited by the field separator character FS (hex 1C) except for the STX and ETX at the start and end of the packet respectively. Below you will see examples of formed packets and responses.

- No field ever contains a non-printing character (< hex 20) *–except for history transactions where subfields of the **Text** field are delimited by GS and RS characters. See* **Special Considerations.**

- No packet ever contains a null (hex 00).

- **Track 1** and **Track 3 data** fields are largely ignored in favor of **Track 2 data** (either mag stripe or key-entered HR – human readable - number). If Track 2 is not present, eGiftSolutions will look at Track 1 field. Track 3 is ignored for processing.

- Key-entered sales are accepted. The card number for key-entered sales shall be inputted in the field of **Track 2 data.**

- All fields are required with a submission unless they are indicated in the tables as “Optional, ignored”, “Reserved”, or “Varies by transaction”. In those cases of “Varies by transaction”, a field might be required for a particular transaction and is noted below. For example, to perform a **Void,** you will need to input the original **Transaction ID** of the transaction you want to voided into the packet.

- Add Value and Activation Transactions may not be voided. For record keeping and cash recording purposes, those transactions involve the real exchange of virtual gift balances for cash or credit/debit card transactions so in order to void (or cancel) a card activation request you must deactivate the card. While the proper procedure to reverse these is up to the business owner, we recommend **Deactivation** in the case of reversing

an Activation and a **Redeem** in the case of reversing an Add Value. As a rule, we feel that a transaction should be processed and cash or credit/debit should be tendered BEFORE any Add Value or Activation transaction. This ensures money is in the till and then a card is duly activated or has value added to it and can ensure proper reporting and record keeping of cash receipts.

ASCII (dialup)

- Data are transmitted in seven-bit ASCII with even parity in the high-order bit position, i.e., the count of '1' bits in a byte is always an even number. For example, STX is transmitted as hex 82 but ETX, which already contains an even number of bits, is transmitted as hex 03.

- The LRC value transmitted as the last character of a packet is the cumulative exclusive-OR of all preceding characters *except the STX,* which does not participate in the calculation.

- When a sent packet is received with the proper character parity and LRC, the receiving station responds with ACK (hex 06) and processed the content. A packet that fails the parity or LRC test receives a NAK (hex 15) and will not be processed. Sender should retry transmission up to six times.

- Unlike HTTPS transmission below, *no character is escaped.*

HTTPS (SSL via TCP/IP)

- Most software, virtual terminal and POS systems communicate over SSL.

- Data are transmitted in the HTTPS protocol. This is distinguished from the ASCII dialup protocol above as follows:

- No character parity is generated or enforced.

- An LRC character is optional but not looked at.

- Every non-printing character (STX, ETX, FS, etc.) is escaped according to HTTPS rules; e.g., STX (hex 02) becomes the string '%02', FS becomes '%1C', etc.

- ACK and NAK are neither generated nor expected, as one relies on the HTTPS protocol to be error-free.

2. Transaction types

Note: For the typical transaction request, the server vets the following fields:

- Card number (extracted from Track data or inputted via key-entry).
- Merchant ID (Overall Merchant Group ID, 7 numbers) – used to identify the Merchant.
- Location ID (Store ID, 4 characters) – used to identify a particular Merchant location. A single store could be one of the many locations of a Merchant.
- Terminal ID (Terminal ID, 4 characters) – used to identify a terminal at a merchant location. A Terminal is a single device used at a merchant location. Terminals can be shared or separated. In other words, 5 terminals can run on Terminal ID 0001 or they can be broken out for reporting purposes of 0001, 0002, 0003, etc. The terminal ID submitted in the packet becomes a combination of MID + LID + TID.

Table of transaction types and input and output fields.

(See separate page containing table.)

Note: Input fields must be delineated by FS (hex 1C) whether present or omitted. The server expects 26 input fields and returns 7 output fields. Signify an omitted field by two consecutive FS characters. (For the final field, the trailing delimiter is ETX, not FS.)

3. Transaction URL for POST

```
https://giftcardserver.com/isapi/gcard.dll
```

Port 443

4. Transaction request (terminal to server)

Field definitions in the order they occur

INPUT

EDC Type	G1
Transaction Code	3 characters
Track 1 Data	Variable length field; null if card number is manually entered.

Track 2 Data	Card # or raw card data
Track 2 Format Code	0 or Blank: Formatted by terminal; Nonzero: Raw track data.
Track 3 Data	Variable length field; null if card number is manually entered.
User PIN	Optional, ignored
Group / Merchant ID	7 digits
Store / Location ID	4 digits
Terminal ID	15 digits
Communication Type	Reserved
Terminal Type	e.g., O3750, Vx570, N8320, N8400, N8010, N2085.
Terminal App Version	e.g., 2.6
Time Zone	e.g., EST, CST
Industry code	1: Retail; 2: Restaurant.
Serial number	e.g., 1008078745
Entry Mode	0: Other; 1: Magnetic; 2: Manual; 3: Barcode; 4: Contact-less.
Rule ID	Optional, ignored
Item No	Optional, ignored
Cashier / Clerk ID	Optional, ignored
Date	Optional, ignored
Time	Optional, ignored
Transaction ID	Varies by transaction
Amount 1	Amount of the **Redeem**, **Add Value**, **Activation** otherwise this field Varies by transaction
Amount 2	Varies by transaction
Text	Varies by transaction

STX (Start of text).

Hex 02 (ASCII) or the string '%02' (HTTPS).

26 data fields, delimited by FS.

EDC Type.

'G1'.

Transaction code.

3 digits zero-filled. *The current server version accepts any number of digits here.*

Track 1 data.

Optional under Common Considerations (see note above).

Track 2 data.

Key-entered eGiftSolutions HR (human readable) card number from back of card. Also swiped eGiftSolutions-issued card number, optionally followed by the '=' character and additional information which the server discards. *Or, if the Track 2 format code field is nonzero, raw data from card swipe which the server is expected to parse. The server expects up to 50 characters but will not enforce that limit if the terminal happens to exceed it.*

Track 2 format code.

0 or omitted: Formatted by terminal; Nonzero: Raw data.

Track 3 data.

Optional under Common Considerations (see note above).

User PIN.

Reserved for future use.

Group ID (Merchant ID).

7 digits zero-filled, e.g. 0001234

Store ID (Location ID).

4 digits zero-filled, e.g. 0001

Terminal ID.

15 digits zero-filled. The first 11 digits duplicate the group and store IDs.
Example: 000123400010001

Merchant 0001234

Location 0001

Terminal 0001

Communication type.

Reserved.

Terminal type.

e.g. e.g., O3750, Vx570, N8320, N8400, N8010, N2085, Micros, PCAmerica, Datacap, Dinerware, etc.

Terminal app version.

e.g., 2.6.

Time zone.

e.g., EST, CST. *Use reserved for future implementation.*

Industry code.

1: Retail; 2: Restaurant.

Terminal serial number.

e.g., 1008078745.

Entry mode.

0: Other; 1: Magnetic (swiped); 2: Manual (key-entered); 3: Barcode (scanned); 4: Contact-less (RFID).

Rule ID.

Reserved for future use.

Item number.

Reserved for future use.

Cashier ID.

Reserved for future use.

Date.

Reserved for future use.

Time.

Reserved for future use.

Transaction ID.

Reserved for future use.

Amount 1.

Varies by transaction type.

Amount 2.

Varies by transaction type.

Text.

Varies by transaction type.

ETX (End of text).

Hex 03 (ASCII) or the string '%03' (HTTPS).

LRC (ASCII only, ignored in HTTPS).

5. Transaction response (server to terminal)

Field definitions in the order of response packet

OUTPUT

Result Code	501: Info; 502: Success; 503: Declined; 902 Voided by Host
Transaction ID	6 characters
Rule ID	Reserved, null
Amount 1	Amount of transaction
Amount 2	Card balance after transaction
Decline Reason	See table
Text	Varies by transaction

STX (Start of text).

Hex 02 (ASCII) or the string '%02' (HTTPS).

Precisely 7 data fields, delimited by FS.

Result code.

See Result Code Table in 11. Result Code of Decline/Error would correspond to the Decline/Error Code Table.

Transaction ID.

A six-digit pseudorandom number.

Rule ID.

Reserved for future use.

Amount 1.

Amount of transaction.

Amount 2.

Card balance after transaction. This can be returned onto a receipt, email or screen for display.

Decline reason code.

See Decline Reason Code Table in 10. For successful transactions, the Decline Code would be returned as '000'.

Text.

ETX (End of text).

Hex 03 (ASCII) or the string '%03' (HTTPS).

LRC (ASCII only, ignored in HTTPS).

6. Special considerations – Terminal based integrations

There are certain types of transactions that can be used for returning card history as well as communicating with eGiftSolutions tech and customer service departments for ordering cards and paper. The latter transactions are typically used for terminal processing. In the case of a History transaction, you would input the 'From' and 'To' dates separated by a RS in the Text field of the request packet. The output would be formatted and you can print or display on a screen.

Note: These transactions are not applicable to most API integrations as reporting is available online and the customer service transactions are used more in terminal-based applications.

Card History Transaction

Input

The input **Text** field expected by the server contains the following values:

'From' date.

YYYYMMDD

RS (Record separator).

Hex 1E (ASCII) or '%1E' (HTTPS).

'To' date.

YYYYMMDD

Output

The output **Text** field returned by the server contains the following values:

Number of Transactions (*n*) in current packet.

Range 1 to 10.

RS (Record separator).

Hex 1E (ASCII) or '%1E' (HTTPS).

n ***occurrences of the following group.***

Date YYYYMMDD.

RS.

Time HHMMSS.

RS.

Transaction ID.

RS.

Transaction Amount in cents.

GS (Group Separator).

Hex 1D (ASCII) or '%1D' (HTTPS).

Not present in final packet; immediately followed by ETX instead.

Customer Service Requests

Input

The input **Text** field expected by the server contains one of the following values (one ASCII digit):

1 (Help Desk Request)
2 (Paper Order)
3 (Internal; do not use)
4 (Order Cards)

7. Sample Transaction Format

Activate Card Request

Note: This is how your data should look when converted for HTML submission

```
%02G1%1C005%1CB6265555707036131%5E%5E0000%1C6265555707
036131%3D0000%1C1%1C%1C%1C0000001%1C0001%1C00000010001
0001%1C2%1CVX520%1C3%2E8%1C%1C1%1C286%2D334%2D575%1C1%
1C0001%1C00001%1C0001%1C20171023%1C0959%1C%1C1000%1C%1
C%03
```

Exploded

%02

G1 (EDC Type)

%1C

005 (Transaction Code for Activation)

%1C

B**6265555707036131**%5E%5E0000 (Track 1 Data – Card # in Bold for example)

%1C

6265555707036131%3D0000 (Track 2 Data – Card # in Bold for example)

%1C

1 (Format Code)

%1C (FS before Track 3 Data – not included in packet)

%1C (FS before User Pin – not included in packet)

%1C

0000001 (MID)

%1C

0001 (LID)

%1C

000000100010001 (TID – The TID = MID + LID + TID)

%1C

2 (Communication Type)

%1C

VX520 (Terminal Type – Terminal Type for integration is provided to the end-user by eGiftSolutions during certification process and remains constant for that integration’s use.)

%1C

3%2E8 (App Version – in this case 3.8 - %2E hex is a .)

%1C (FS before Time Zone – not included in packet)

%1C

1 (Industry Code)

%1C

286%2D334%2D575 (Serial Number, ID Number or MAC Address of Device Communicating with eGiftSolutions provided by software/hardware)

%1C

1 (Entry Mode)

%1C

0001 (Rule ID)

%1C

00001 (Item #)

%1C

0001 (Clerk ID/Cashier ID)

%1C

20171023 (Date)

%1C

0959 (Time)
%1C (FS before Transaction ID – left blank in this example)

%1C

1000 (Amount – 10.00 without decimal or symbols)

%1C (FS before Amount 2 – left blank)

%1C (FS before Text – left blank)

%03 (End of String)

Response

%02502%1C802773%1C0001%1C000001000%1C000001000%1C000%1CCard%20***6131%20Activated%20for%20%2410.00%03

Exploded

%02

502 (Result Code)

```
%1C

802773 (Transaction ID)

%1C

0001 (Rule ID)

%1C

000001000 (Amount of transaction without decimal)

%1C

000001000 (Balance after transaction – can be reported
on the screen and/or receipt)

%1C

000 (Decline Reason – if applicable)

%1C

Card%20***6131%20Activated%20for%20%2410.00%03

(Response Text)
```

8. Transaction Types

Transaction Type #	Transaction Type Description
001	Card Balance Request
002	Redeem Request
003	Add Value Request
004	Void
005	Card Activation Request
006	Card De-activation w/ refund
007	Card De-activation w/o refund
008	Card Re-issue Request
009	Store Credit Request

012	Tip Adjust
014	Balance Transfer
020	Add points
021	Redeem points
022	Expiration Date Adjust
023	Point balance inquiry
024	Activate points *(Future Use)*
025	Point refund *(Future Use)*
026	Point balance transfer *(Future Use)*
027	Void points *(Future Use)*
200	Card History Request – Requires “Text” Field input; used for IP Terminal implementations mostly
210	Point history *(Future Use)*
301	Sign in (with or without a PIN) *(Future Use)*
302	302 - Sign out (require reason code) *(Future Use)*
303	Change PIN *(Future Use)*
010	Sign-On *(Future Use)*
011	Sign-Off *(Future Use)*
118	Customer Service – Requires “Text” Field input; used for Terminal implementations mostly

9. Transaction functional description

For most transaction types, the submission of 26 fields with the unique identifier of **Transaction Code** is the only requirement. There are few transactions such as **Void** that

require a prior Transaction ID or other information. Those instances are noted below and the in the **2009-12-21-Message Formats (G1)_API_Companion.xls.**

Balance Inquiry – 001

Returns dollar and point balance for any of the merchant's eGiftSolutions cards.

Redeem – 002

Deducts transaction amount, or remaining balance, whichever is smaller, from stored card balance. Returns balance left in response packet. You can adjust a tip on the fly (aka Counter Tip or Tip in Transaction) by inputting the primary amount in Amount 1 and the Tip Amount in Amount 2.

Add value – 003

Adds transaction amount to stored card balance. If card is already active, transaction is possible. Otherwise, they must perform an Activation first.

Void Transaction – 004

Reverses specific transaction. Internally flags it voided so it cannot be voided more than once. **Transaction ID** of the original transaction is required to **Void.** That is put in the **Text** field for processing. You may not void Add Value or Activation transactions.

Activate card – 005

Like **Add Value** but also creates new card record in database. Can only be done on a card that has never been activated. If card already exists, use Add Value transaction.

Deactivate Card (with refund) – 006

Changes card status to inactive, zeroes balance.

Deactivate Card (no refund) – 007

Changes card status to inactive, balance unchanged. The amount of the balance is returned so a refund can be given. Card remains deactivated.

Reissue card – 008

Like **Deactivate with refund** but also creates new card record in database and transfers balance to the new card. The second card goes in the **Text** field.

Store credit – 009

Functionally same as **Add Value/Activation.** If you select the Store Credit function, you will activate a new card for the amount of the store credit. This function does not record as an Add Value or Activation which assumes funds are

brought into the business, but records as a Store Credit which assumes goods were returned in exchange for a merchandise credit on a gift card. This can also be used for donations by a business where no money is taken in.

Tip Adjust – 012

Modifies a specific transaction updating tip and total. Requires the Transaction ID of the original transaction to adjust plus the Tip Amount in Amount 2 Field.

Balance transfer – 014

Like **Reissue card** but receiving card already exists. The second card receiving the balance goes in the **Text** field. Client may want to key in the first number and swipe the second card when a card's magnetic strip is damaged to transfer balance from one card to a new card.

Add points – 020

Adds transaction amount to stored card point total. *Loyalty implementation only.*

Redeem points – 021

Deducts transaction amount from stored point total. *Loyalty implementation only.*

Expiration Date Adjust – 022

Modifies a specific card updating expiration date to default or specified value. *Future implementation.*

Point Balance Inquiry – 023

Returns point total only. *Loyalty implementation only.*

Customer Service Functions – Terminal Implementation Only – 118

Initiated by merchant.

Help desk

Generates email to eGiftSolutions Coordinator for order processing.

Order paper

Generates email to eGiftSolutions Coordinator for order processing.

Merchant signup

Redirected from **Activate Card** code under special conditions when a 'Merchant Activation' card is swiped. Creates merchant and location ID. Also creates dummy activator card record so activation can't be used more than once.

Order cards

Generates email to eGiftSolutions Coordinator for order processing.

History Transaction – 200

Returns card history for specific date range. Results in multiple data packets requiring special handling. Terminal prints a history report using these packets. Due to necessary bandwidth, this feature is unavailable in dial-up format.

10. Decline/Error Message Codes

Error #	Error Message	Error Meaning
000	No Error	Transaction Approved
001	Invalid Account Id	Operation Failed. No such card
002	Account Not Active	Operation Failed. Card not active
003	Invalid Currency	Operation Failed. Wrong Currency
004	Invalid Account Type	Cannot operate in this Group (wrong Group ID)
005	Invalid Vendor Id	Cannot operate in this Store (wrong store ID)
006	Empty Password	Operation Failed. Missing Password
007	Password Not Correct	Operation Failed. Wrong Password. Try Again
008	Too Many Wong Passwords	Operation Failed. Wrong Passwords, Call for Support.
009	Card Already Active	Operation Failed. Card Already Active
010	Transaction Already Canceled	Operation Failed. Transaction Already Canceled
011	Transaction Not Found	Transaction ID not valid
012	Card Expired	Operation Failed. Card has expired
013	Server Error	Call Merchant support

014	Card Cannot be Activated	Bad Check Digit
015	Invalid Terminal ID	Terminal ID not in the database
016	Malformed Request	Request packet is malformed; Declined
100	No Funds; Declined	No Funds on Card; Card Declined
101	Insufficient Funds	Allow transaction, set balance to zero, and terminal will provide split payment feature.

11. Result Message Codes

Result Code	Result Message	Result Meaning
501	Informational Response Successful – Non Monetary	Transaction Request Approved – This response would come from the requests for Customer Service Request or Balance Inquiry Transactions, e.g. Your Paper Order Was Received and is Non-Monetary in nature
502	Transaction Successful	The particular transaction request submitted has been successful
503	Declined	See Decline Reason Codes Table
902	Voided Transaction	Transaction has been voided by server