Standard POS device message specifications manual

ACI™

trusted globally

payment systems

© 2011 by ACI Worldwide, Inc. All rights reserved.

All information contained in this document is confidential and proprietary to ACI Worldwide, Inc., or one of its subsidiaries. This material is a trade secret and its confidentiality is strictly maintained. Use of any copyright notice does not imply unrestricted or public access to these materials. No part of this document may be photocopied, electronically transferred, modified, or reproduced in any manner without the prior written consent of ACI Worldwide, Inc., or one of its subsidiaries.

NEITHER ACI WORLDWIDE, INC., OR ITS SUBSIDIARIES SHALL BE LIABLE FOR ANY DIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, WHETHER RESULTING FROM BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, OR OTHERWISE, EVEN IF ACI WORLDWIDE, INC., OR ITS SUBSIDIARIES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. ACI WORLDWIDE, INC. RESERVES THE RIGHT TO MAKE CHANGES TO THE INFORMATION CONTAINED HEREIN AT ANY TIME WITHOUT NOTICE.

Contents

Optional Data Subfields — FID 6 *continued*

Download Data *continued*

ACI Worldwide, Inc.

What's New

Here is what's new in the ***ACI Standard POS Device Message Specifications Manual*** since it was published for release 6.0, version 10.

June 2011

This table highlights the updates made to the ***ACI Standard POS Device Message Specifications Manual*** (release 6.0, version 10) in June 2011. The first column of the table lists the sections and appendixes in which the changes were made; the second column describes the changes.

Section/ Appendix	Major Changes
All	Changes the name of "EMV Additional Response Data" for FID 6, subFID R to "EMV Reversal Data/EMV Additional Response Data" throughout the manual to better reflect the subFID's usage.
2	Updates the summary table for FID n — Totals/Employee to indicate that it is not used in request messages. Updates the summary table for FID 6 subFID R — EMV Reversal Data/EMV Additional Response Data to indicate that it is used in request messages. Corrects the Picture clause in the summary table for FID 6 subFID R — EMV Reversal Data/EMV Additional Response Data in request messages. For requests, it should be PIC X(82). Updates the summary table for FID 6 subFID h — System Trace Audit Number (STAN) and FID 6 subFID i — Retrieval Reference Number to indicate that they are used in responses and not in requests. Updates the Card Type description for FID 6, subFID D — Purchasing Card or Fleet Card Data to include the valid value A (American Express purchasing card).

ACI Worldwide, Inc.

Preface

The ***ACI Standard POS Device Message Specifications Manual*** describes the ACI standard POS message exchanged between a host and third-generation POS devices or controllers. This manual describes the transactions supported by the ACI standard POS message, the format of each data element that can be included in transaction request and response messages, and descriptions of the message flows between the host and POS devices. It also discusses the common framework and infrastructure to support the communication of these messages and the processing options that need to be considered by the host customer or service provider and the POS device hardware vendor when implementing them.

Audience

This manual is intended as a reference for persons responsible for designing and coding the message interface between a host and the POS hardware device. It is intended for both retail customers or service providers implementing the ACI standard POS message at a host and for POS terminal vendors implementing the ACI standard POS message in terminal firmware.

Prerequisites

Readers of this manual should be familiar with standard retail point-of-sale business practices and terminology. A strong understanding of software programming and data communications is also recommended.

Additional Documentation

This manual contains references to the following American National Standards Institute (ANSI) and International Organization for Standardization (ISO) publications:

- The ANSI X9.19 (1986) standard, ***Financial Institution Retail Message Authentication Standard***, describes the use of message authentication codes (MACs).

- Currency codes used in the ACI standard POS message are based on the ISO 4217 standard, ***Codes for the Representation of Currencies and Funds***.
- Country codes used in the ACI standard POS message are based on the ISO 3166 standard, ***Codes for the Representation of Names of Countries***.
- Cryptogram transaction types are represented by the first two digits of the processing code from the 1987 ISO 8583 standard, ***Bank Card Originated Messages—Interchange Message Specifications—Content for Financial Transactions***.
- Track 1 data is provided in the format specified in the ISO 7813 standard, ***Identification Cards—Financial Transaction Cards***.

Software

This manual documents standard processing as of its publication date. Software that is not current and custom software modifications (CSMs) may result in processing that differs from the material presented in this manual. The customer is responsible for identifying and noting these changes.

Manual Summary

The following is a summary of the contents of this manual.

"Conventions Used in This Manual" follows this preface and describes notation and documentation conventions necessary to understand the information in the manual.

Section 1, "Introduction," introduces the ACI standard POS message and the terminals, protocols, features, transactions, and message types it supports.

Section 2, "The ACI Standard POS Message," describes the standard message header fields, optional data fields, and header and optional data field requirements for request and response messages. Examples are provided for standard header and optional data field settings in various request and response messages.

Section 3, "Download Data," describes the data elements that can be downloaded from the host to a POS device in full or partial downloads. It explains full and partial download processing in detail and provides message flow examples.

Section 4, "Configuration Considerations," discusses the following processing issues that impact how the ACI standard POS message is implemented: configurable receipts, draft capture, message sequencing, transaction accumulation totals, PIN encryption, message authentication codes (MACs), handshaking, and mail support.

Section 5, "Exception Transaction Message Flows," contains examples of message flows between a POS device and the host for a variety of exception scenarios.

Appendix A, "Category Code Examples," explains how the Category Code field is used in download processing. This field is contained in the first two bytes of FID V (Mail/Download Key).

Appendix B, "American Express Standard Industry Formats," describes the formats of the AMEX Data Collection field (FID 0) for draft capture of transactions originating from American Express cardholders.

Publication Identification

Three entries appearing at the bottom of each page uniquely identify this ACI publication. The publication number (for example, PS-DP140-10 for the ***ACI Standard POS Device Message Specifications Manual***) appears on every page to assist readers in identifying the manual from which a page of information was printed. The publication date (for example, Jun-2011 for June, 2011) indicates the issue of the manual. The software release information (for example, R6.0v9 for release 6.0, version 10) specifies the software that the manual describes. This information matches the document information on the copyright page of the manual.

ACI Worldwide, Inc.

Conventions Used in This Manual

This section explains how optional data fields, subfields, and blank characters are documented in this manual.

Optional Data Field and Subfield Description Format

Optional data fields and optional data subfields are described in section 2 using a standard format. Each field and subfield description begins with a title structured as shown below.

FID x — Field Name
SFID x — Subfield Name

FID *x*	The field identifier for this field, where *x* is the alphabetic (A–Z, a–z) or numeric (0–9) character that identifies the field.
Field Name	A text description of the data associated with the FID.
SFID *x*	The subfield identifier for this subfield, where *x* is the alphabetic (A–Z, a–z) character that identifies the subfield.
Subfield Name	A text description of the data associated with the SFID.

The FID/SFID title is followed by these standard labeled fields.

Request: Indicates whether the FID/SFID can be included in request messages. *Optional* indicates that the customer can configure the FID/SFID to be included in request messages. *Not available* indicates that the FID/SFID cannot be included in request messages. An indication of whether the request FID/SFID length is fixed or variable is also included.

Response: Indicates whether the FID/SFID can be included in response messages. *Optional* indicates that the customer can configure the FID/SFID to be included in response messages. *Not available* indicates that the FID/SFID cannot be included in

responses. FIDs/SFIDs that are echoed from the request are also noted. However, even FIDs/SFIDs that are noted as being echoed in responses are not included in responses at all, unless they are configured to be included at the host. An indication of whether the response FID/SFID length is fixed or variable is also included.

The standard labeled fields are followed by a description of the FID/SFID and its use. The description includes the valid values associated with the FID/SFID, if applicable, and if the data associated with the FID/SFID consists of a group of fields, these group fields are described as well.

Blank Characters

Field descriptions for internal and external messages, tokens, files maintenance screens, and reports often include lists of valid values. When the value contains a blank character, this manual uses the symbol ␣ to indicate the blank character.

1: Introduction

The ACI standard point-of-sale (POS) message provides complete and flexible support for third-generation POS devices and for controllers from a customer host. The ACI standard POS message has been developed through years of research and experience in both domestic and international markets in conjunction with major POS device vendors. The flexibility of the ACI standard POS message allows retail customers to work with any device vendor and select the transactions and functions appropriate to their individual retail environments.

This section introduces the ACI standard POS message and the terminals, features, transactions, and message types it supports.

Throughout this manual, the term *host* refers to a point-of-sale authorization and switching system.

The ACI Standard POS Message

The ACI standard POS message defines the structure of request and response messages exchanged between a host and ACI standard POS message-compliant POS devices. Each message is comprised of a mandatory standard message header followed by any number of optional data fields. Each message is surrounded by the data communication characters appropriate to the communications protocol implemented. Every message sent between the terminal and the host must contain the standard message header. The header contains information about the terminal, the transaction, and, optionally, the employee who initiated the transaction.

Although the standard header can stand alone, any number of optional data fields can follow it. The only restrictions on the number of optional data fields that can be included in the message are the protocol limitations, the size of the terminal read buffer, and the size of the host read buffer. The host read buffer is configurable to a maximum of 4,088 bytes, including all data communications control characters.

Each optional data field included in an ACI standard POS message is identified with a one-character *field identifier*, or FID, immediately preceding the optional data. Each FID is prefaced with a *field separator* character. Some optional data fields contain subfields that are identified by a one-character *subfield identifier*, or SFID, immediately preceding the optional subfield data. Each SFID is prefaced with a *record separator* character.

Data elements that are downloaded to POS devices are identified with a one-character *download field identifier*, or DID, preceding the data element. All DIDs are transmitted in Mail Text/Download Data (FID W) in download responses from the host to the terminal. A total of 82 data elements can be included within the format, although only 68 are currently supported (14 DIDs are reserved for future use). A field separator character is used to identify where FID W begins and ends. Within FID W, every data element included is prefixed with a *group separator* character, a DID character, and a space.

All data carried in an ACI standard POS message, excluding data communications control characters, must be in ASCII format. Any binary coded data used at the POS device must be converted into ASCII before being included in a request message to the host. This conversion must be reversed to retrieve binary coded data from an ASCII response received from the host.

The standard message header and each of the optional data fields and subfields are described in detail in section 2 of this manual. Downloadable data elements are described in detail in section 3 of this manual.

Supported Terminals

Multiple terminal vendors have worked with the ACI standard POS message as set forth in this manual. Processors and retailers have used the specifications in developing the POS device functionality that meets their POS EFT requirements.

The ACI standard POS message is flexible enough to be used with any sophisticated third-generation POS device having the capability of sending messages to and receiving messages from the host using the message formats supported by the host. The ACI standard POS message can be used between the host and compliant POS devices or between the host and compliant store controllers. Customers can use any POS device or controller that conforms to the requirements set forth in this manual.

Because of the flexible nature of the ACI standard POS message, customers and vendors must collaborate on the transactions to be supported, and the optional elements to be used in those transactions.

Supported Features

An enhanced set of features is available for use with the ACI standard POS message. Customers can select and configure the features necessary to meet their processing requirements. This subsection provides a list of these features, followed by a summary of each feature. The following features are supported by the ACI standard POS message:

- Expanded transaction set
- Multiple terminal vendor support
- Configurable messages
- Multiple language support
- Configurable receipts
- Download options
- Draft capture options
- American Express data collection
- Settlement and cutover support
- Terminal balancing
- Transaction security
- Derived unique key per transaction (DUKPT) support
- American Express card security codes (CSCs)
- Message sequencing
- Mail support
- PS2000 support
- Track 1 and Track 2 processing
- Address verification
- Stored value cards
- Balance inquiry service
- Electronic check authorization
- Electronic benefits transfer (EBT) support

- Europay, MasterCard, and Visa (EMV) chip card support
- Multiple currency support
- Contactless Transaction Support
- Dynamic Card Verification Values
- Healthcare/Transit Auto-Substantiation Transactions
- Healthcare Eligibility Inquiry Transactions
- Visa Card Level Results

Expanded Transaction Set

The ACI standard POS message can handle processing for a total of 33 different POS transactions sent from the terminal, including both financial and administrative transactions. Customer flexibility and functionality are greatly increased because the ACI standard POS message supports a wide variety of transaction types. Some of the transactions supported include check guarantee, check verification, purchase with cash back, preauthorization purchase, preauthorization purchase completion, and request mail. A complete list of the transaction set supported is included later in this section.

Multiple Terminal Vendor Support

Customers can use any POS terminal or controller that conforms to the ACI standard POS message requirements set forth in this manual. The ACI standard POS message is flexible enough to communicate with any sophisticated third-generation POS terminal having the capability of sending messages to and receiving messages from the host.

Configurable Messages

The ACI standard POS message allows customers to configure message requests and responses according to their needs. The host can read requests from the terminal that include any number of optional data fields. These optional data fields can be in the request in any order. The data fields included in responses sent to the terminal can also be configured by customers.

Customers can select a different message format for each transaction type, if desired. The request message and the response message for each transaction type can consist of different fields. The fields included in the request message can be in any order the customer chooses.

All ACI standard POS messages consist of a standard header and optional data fields. The standard header is mandatory and can stand alone. For example, in Logon, Logoff, and Handshake transactions, no optional data fields are required. For other transactions, however, the standard header can be followed by any number of optional data fields. Optional data fields are subject to the requirements of the transaction. This means that certain transactions may require the presence of certain data fields in their messages. The only other restrictions on the number of optional data fields that can be included in the message are the protocol limitations, the size of the terminal read buffer, and the size of the host read buffer. The host read buffer is configurable to a maximum of 4,088 bytes, including all data communication characters associated with the message.

Multiple Language Support

Customers have the option of formatting display responses to a terminal in one of three different languages. The terminal can determine which language to use based on information contained in its request to the host. The host uses information contained in the terminal request to access language tables configured by the customer, or it uses information contained in a database to determine the language.

If no language is specified in the request, terminal responses are displayed in the default language set up for the terminal in a host database. If a language table is specified in the request, the information in language tables configured by the customer override the default value set up in the database. In short, the language table information must be requested by the terminal in order to be used. Otherwise, the default value in the database is used.

Configurable Receipts

Using the ACI standard POS message, the host does not format receipts. Customers must determine if receipts are necessary and then supply the optional data using a host database. It is the customer's responsibility to configure each response in the terminal configuration data to include sufficient data for the terminal to format and print a receipt. The host has the capability of returning a response determined by the customer in up to three different languages as well as returning a 48-character response.

Download Options

The ACI standard POS message supports both full and partial downloads. The terminal can receive full configuration file information or single records. Whether a full or partial download is requested is determined at the terminal level. Download data is flexible, allowing customers the ability to send more than 1,000 bytes of discretionary processing data to the terminal. In addition, the download data can pertain to as many terminals as the customer selects. A separate download record is **not** required for each terminal.

The ACI standard POS message allows the following information to be downloaded to terminals:

- PIN, MAC, and data encryption keys
- PIN pad character
- Allowed transactions
- Floor limits
- Return and adjustment count and amount limits
- Information for up to 30 card prefixes
- Referral phone number
- Service representative information
- Miscellaneous terminal information
- Any information the terminal owner and operator want to include or that the terminal vendor requires

Draft Capture Options

The ACI standard POS message provides an efficient method of draft capture with third-generation terminals by performing authorization and draft capture in a single transaction. If a transaction is not completed correctly, the operator who entered the transaction sends an adjustment transaction.

The ACI standard POS message also supports transactions that require paper follow-up. This method consists of authorizing a transaction online and then sending a paper voucher/draft follow-up. This method requires merchants to submit the paper voucher/draft follow-up in order to receive payment.

American Express Data Collection

American Express has defined categories into which transactions being processed are placed. For each of these categories, American Express requires different data to be sent from the device. To ensure the data they require is captured by the device and sent to the host, American Express has set forth standard industry formats for each category. In order to perform the collection of data on transactions originating from an American Express cardholder, customers must use messages flexible enough to accept and recognize the American Express standard industry formats. The ACI standard POS message supports these standard industry formats, therefore, allowing the host to capture and process transactions using them.

For the American Express standard industry formats, refer to appendix B.

Settlement and Cutover

Settlement, in terms of the host, is actually a reconciliation effort whereby the transaction activity for the current day is closed out and reported, and the system is shown to be in balance. The term *cutover* refers to the point in time where transactions stop being logged to the current day's business and start being logged to the next day's business. It refers to the closing of the current posting day and the opening of the next.

Host settlement differs from the settlement carried out by participating financial entities (i.e., retailers, retailer sponsors, cardholders, card issuers, and interchanges) in that the host does not move funds to settle accounts. It does, however, provide the support that allows these financial entities to initiate the movement of funds.

The ACI standard POS message provides the following administrative transactions, which allow customers to effectively close out business totals in accordance with retailer needs:

- Logon and Logoff transactions
- Close Batch, Shift, and Day transactions

Terminal Balancing

The ACI standard POS message enables merchants to balance their terminals with the host. The host retains terminal totals and enables the retailer to request totals information from the host in order to balance their totals and receipts against the host. Administrative transactions enable merchants to request subtotals for an employee, batch, shift, or day.

The host maintains totals for each terminal at a merchant site. Additionally, the host can maintain a set of totals that combines all of the terminals at a merchant site. Terminals in a department store or a pay-at-the-pump service station are examples of site configurations.

Merchants have at least two methods for merchants to balance terminals with the host. Merchants have the option of performing subtotal request transactions to determine if the host and the terminal have accumulated the same totals. Merchants then have the option of entering adjustment transactions, if necessary, before closing the batch.

Merchants also have the option of balancing terminal and host totals after each transaction is performed. This method can be accomplished by configuring the host to return subtotals with each response message.

In addition, merchants also can receive a report a day later, or contact the service provider to receive terminal balancing information.

Transaction Security

The ACI standard POS message offers various ways of ensuring that transactions are transmitted and secured correctly. These include options that ensure every transaction is processed only once and options that allow customers to select from various PIN encryption methods.

The ACI standard POS message supports the use of message authentication codes (MACs). MACs are used to verify whether data has been accidentally or fraudulently altered. Any data sent between the terminal and the host can be verified through the use of MACs. The customer can decide whether to use MAC verification with a particular transaction by configuring MAC parameters at the host.

The ACI standard POS message provides data encryption for the Available Balance field.

The ACI standard POS message supports full message encryption and configurable message encryption.

The ACI standard POS message supports dynamic key management, which is the automatic replacement of working keys for a terminal based on one or more thresholds, for the following working keys:

- The PIN communications key (KPE).
- The MAC communications key (KMAC).
- The data encryption communications key (KME).

The transaction acquirer can determine which PIN encryption, message authentication, and data encryption keys are to be used in a transaction. The host includes key indexes in requests to the host security module, and can return the indexes to interchanges that require them. The host supports the following formats for PIN encryption and message authentication keys:

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

The ACI standard POS message supports full message encryption and configurable message encryption. Full message encryption allows the institution to encrypt all optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. Configurable message encryption allows the institution to encrypt specific optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. The specific optional data fields are configured to be encrypted in the ACI Standard Device Configuration File (ACNF) request and response field maps. Full message encryption and configurable message encryption are enabled through a setting in the POS Terminal Data File.

Setting up transaction security is explained in section 4.

Derived Unique Key per Transaction (DUKPT) Support

The ACI standard POS message provides support for POS devices that derive a unique key for each transaction processed. Each unique key is derived from a base derivation key loaded into the terminal's security module (TSM) when it is initialized. The same base derivation key is loaded into a database at the host. When a POS device derives a unique key for each transaction, the PIN block and MAC in the request message is encrypted under the derived key and the request

message includes a key serial number field in the message. The key serial number consists of a host derivation key identifier, a host derivation key subordinate identifier, the TSM identifier, and a transaction counter. When the host receives a PIN block or MAC encrypted under a derived key, it uses the key serial number passed in the message and the terminal's base derivation key stored in the host database to derive the same unique transaction key, decrypt the PIN block, verify the PIN or MAC, and generate a MAC.

Note: The host may need to translate the PIN block to a single-length master/session key management type PIN block using an intermediate PIN block encryption key before deriving the unique transaction key.

American Express Card Security Codes (CSCs)

The ACI standard POS message contains data that enables hosts to verify American Express card security codes. The three types of card security codes (CSCs) are as follows:

- Three-digit CSC located on the signature panel
- Four-digit CSC located on the front of the card
- Five-digit CSC located on the magnetic stripe

Message Sequencing

The ACI standard POS message supports message sequencing, a method of ensuring that every message being sent from the terminal is received only once and in the correct order. There are two options available for message sequencing. One option calls for the host to check the transmission number, which is a field in the standard message header, in order to detect and drop duplicate requests. The other option consists of comparing message sequence numbers to positively identify the sequencing of a request. If message sequence numbers are included in the message, the host can ensure that no messages have been lost or transmitted out of sequence. The message sequence number method is more reliable, but requires an additional sequence number field to be included in the message itself. For more information on the sequence number field, refer to section 2.

Mail Support

The ACI standard POS message enables host processors to send informational messages to POS terminals throughout a network. Mail messages can also be transmitted from POS terminals to the host processor.

PS2000 Support

The ACI standard POS message is compatible with Visa Payment Service 2000 (PS2000). PS2000 is a Visa program that ensures that the portion of the transaction that is authorized is the same as the portion that is cleared and settled. All phases of the transaction are linked, allowing the institution to reduce back office expenses. The ACI standard POS message supports the PS2000 program for the passenger transport, hotel, automobile rental, direct marketing, and fuel market segments.

Track 1 and Track 2 Support

The ACI standard POS message supports both Track 1 and Track 2 information from the POS device. Refer to section 2 for more information on the data fields used to format Track 1 and Track 2 data.

Address Verification

The ACI standard POS message supports address verification. Address verification assists merchants in controlling losses that can result from credit card fraud during transactions where the cardholder cannot be readily identified. See section 2 for information about the data fields used to format address verification data.

Stored Value Card Support

The ACI standard POS message provides the following transactions for support of stored value cards:

- Card activation
- Additional card activation

- Replenishment
- Full redemption

These transactions enable merchants to implement electronic gift programs.

Balance Inquiry Service

The host can be configured to return balances on Visa Prepaid purchase requests and balance inquiry requests. The purchase request function returns balance information with purchase, purchase with cash back, and preauthorization purchase transactions. The balance inquiry function enables cardholders to request account balance or available credit amounts prior to initiating a purchase transaction.

Electronic Check Authorization

The ACI standard POS message enables POS issuers (or drawee banks) to verify funds availability for check verification and check guarantee transactions based on a transit routing and account number rather than a card number. The customer's account balance is impacted (reduced) for check guarantee transactions. This functionality enables retailers to authorize checks electronically.

Electronic Benefits Transfer (EBT) Support

The ACI standard POS message enables POS acquirers to acquire and route electronic benefits transfer (EBT) transactions to an EBT processor or gateway service provider. This feature enables certain financial transactions to be performed using food stamp or cash benefit accounts.

Europay, MasterCard, and Visa (EMV) Chip Card Support

The ACI standard POS message enables the following transactions to be entered at a POS device using a Europay, MasterCard, and Visa (EMV) chip card:

- Normal purchase
- Preauthorization purchase
- Preauthorization purchase completion

- Mail or telephone order
- Merchandise return
- Cash advance
- Card verification
- Balance inquiry
- Purchase with cash back
- Purchase adjustment
- Merchandise return adjustment
- Cash advance adjustment
- Cash back adjustment

The ACI standard POS message provides support for both cryptogram authentication and script processing for each of the above transactions.

EMV chip cards have both a microprocessor chip and a magnetic stripe. EMV chip cards are integrated circuit cards. An *integrated circuit card* (ICC) is a plastic card, usually the size of a credit card, that contains an embedded microprocessor chip. This chip is capable of storing large amounts of cardholder information and can contain multiple applications. The terms chip card and smart card are sometimes used interchangeably with integrated circuit card.

The ACI standard POS message supports processing for both ICC and magnetic stripe transactions from a single terminal.

The ACI standard POS message supports EMV log-only transactions. The terminal uploads EMV log-only transactions one at a time to the host. These are approved transactions that were authorized offline by the terminal/EMV card. Declined transactions or transactions that were authorized online should not be uploaded. EMV log-only transactions have the same format as online transactions, with all of the usual header fields, FIDs and subFIDs present.

EMV log-only transactions contain a Transaction Certificate (TC), a card-generated cryptogram that provides information about the transaction which can be used if the transaction is disputed.

Multiple Currency Support

The ACI standard POS message provides support for transactions in any defined currency at the POS device. The Transaction Currency Code field indicates the currency in which the transaction is performed at the device.

Contactless Transactions

The ACI standard POS message allows for identifying contactless transactions from terminals. Contactless transactions are transactions initiated without physical contact between the card and terminal.

To identify a transaction as contactless, the terminal must include a value of 07 (contactless chip card transaction) or 91 (contactless magnetic stripe transaction) in the FID 6 subFID E (POS Entry Mode) field in the transaction request.

EMV terminal capabilities are determined from Processing Flag 2 field in the standard message header. If the terminal is capable of contactless EMV transactions, the Processing Flag 2 field must be set to 5.

Dynamic Card Verification Values

The ACI standard POS message supports the use of Mastercard's dynamic CVC3 card verification value (used for PayPass card verification) and Visa's dCVV card verification value (used for Visa Wave card verification). Applicable card verification values are carried in the Issuer Discretionary Data fields of the Track 1 and Track 2 Data. To support these types of card verification, the terminal must include the following in the track information it sends:

- Card verification value (the CVC3 or dCVV)
- Application Transaction Counter (ATC) – a transaction counter maintained by the application on the integrated chip card.
- Unpredictable Number (UN) – a number generated by the terminal and used to create uniqueness in the transaction (required by MasterCard only).

Healthcare/Transit Auto-Substantiation Transactions

The ACI standard POS message supports healthcare/transit auto-substantiation transactions. Auto-Substantiation is the process of verifying that purchase transactions are for expenses permitted under Internal Revenue Service regulations for Flexible Spending Accounts (FSAs) and Healthcare Reimbursement Arrangements (HRAs). Healthcare auto-substantiation transactions enable employers and their third-party healthcare service providers to approve qualified medical expenses at the point-of-sale for purchases made with FSA and HRA payment cards at participating retailers.

Additionally, participating retailers that sell transit fare media, such as commuter passes, parking passes, and mass transit vouchers and tickets, can also use auto-substantiation transactions to approve purchases made with FSA payment cards.

Partial authorizations are supported for auto-substantiation transactions. Similar to other prepaid or pre-funded cards, it is difficult for consumers to spend the exact amount available in FSA or HRA accounts. Partial authorizations allow consumers to make purchases that exceed the account balance, with the remainder of the purchase paid by other means (credit card, cash, etc.)

Healthcare/transit auto-substantiation transactions are handled as purchases. They are identified as healthcare/transit auto-substantiation transactions by the presence of FID 6 subFID l (Healthcare/Transit Data).

The terminal must set the following fields when formatting a healthcare/transit auto-substantiation transaction request.

FID B (Amount 1)	Must be set to the full transaction amount.
FID 1 (PS2000 Data)	The Market Specific Data ID field must be set to M for healthcare (medical) or T for transit.

FID 6 subFID l (Healthcare/Transit Data)	Fields in the first Additional Amount entry must be set as follows: • Account Type – Set to 00 (unspecified). • Amount Type – Set to 4S (amount healthcare) or 4T (amount transit). • Currency Code – Set to the ISO currency code of the amount. • Amount Sign – Set to C (credit, positive balance). • Amount – Set to the amount of the qualified healthcare or transit purchase (right-justified and zero-filled on the left). Although the Additional Amount table can contain from 1 to 6 entries, only entry 1 should be provided in the request. Entries 2 through 6 should not be included.

The terminal will receive FID B (Amount 1) and FID 6 subFID l (Healthcare/Transit Data) in the healthcare/transit auto-substantiation transaction response. If the issuer approves the healthcare/transit auto-substantiation transaction for the full amount, FID B will contain the original requested amount. If the issuer approves the healthcare/transit auto-substantiation transaction for a partial amount (partial authorization), the response fields will be set as follows:

FID B (Amount 1)	Set to the approved partial amount (less than the original requested amount).
FID 6 subFID l (Healthcare/Transit Data)	Fields in the first Additional Amount table entry will be set as follows: • Amount Type – Set to 57 (original amount). • Amount – Set to the original requested amount. Entries 2 through 6 of the Additional Amount table will not be included in the response.

Entries 2 through 6 of the Additional Amount table in FID 6 subFID l (Healthcare/Transit Data) will not be included in the response.

Healthcare Eligibility Inquiry Transactions

The ACI standard POS message supports healthcare eligibility inquiry transactions, which allow healthcare providers (doctors, dentists, hospitals, etc.) to immediately determine the healthcare insurance coverage of FSA or HRA payment card holders.

Healthcare eligibility inquiry transactions are handled as balance inquiries. They are identified as healthcare eligibility inquiry transactions by the presence of FID 6 subFID m (Healthcare Service Data) in the terminal request.

The terminal must set the following fields when formatting a healthcare eligibility inquiry transaction request.

Transaction Code (Message Header)	Must be set to 07 (balance inquiry).
FID 6 subFID U (Transaction Subtype Data)	The Transaction Subtype field must be set to C002 (healthcare eligibility inquiry). The Acquirer Processing Code and Issuer Processing Code fields must be set to blanks.
FID 6 subFID m (Healthcare Service Data)	The Provider ID must be set to the medical license number of the healthcare provider. The Type Code must be set to the HIPAA code for the healthcare service. The rest of the subfield must be set to blanks.

The terminal may request eligibility information for up to five healthcare services within a single request message. Each entry in the subFID m Service table represents a single healthcare service.

The terminal will receive FID 6 subFID l (Healthcare/Transit Data) and FID 6 subFID m (Healthcare Service Data) in the healthcare eligibility inquiry transaction response. SubFID l will contain from 1 to 6 entries in the Additional Amount table.

The Amount Type field in each entry will contain a value of 3S (amount co-payment). SubFID m will contain from 1 to 5 entries in the Service table.

Visa Card Level Results

The ACI standard POS message can return Visa card level results to the terminal if they are present in transaction responses. Visa card level results are codes identifying the participation (rewards) programs in which the card involved in a transaction is enrolled (e.g., Visa Traditional, Visa Traditional Rewards, Visa Signature, or Visa Infinite). Card level results codes are carried in FID 6 subFID k (Card Level Results).

Transaction Support

All of the transactions supported by the ACI standard POS message must originate with a request from the terminal. A response is sent from the host for each request received unless the host is unable to respond. In this case, no response is sent and the transaction times out at the terminal. Message requests and responses are explained in more detail later.

The ACI standard POS message supports both financial and administrative transactions. Financial transactions are monetary transactions performed on behalf of a cardholder, and adjustments to these transactions. Administrative transactions are monetary and non-monetary transactions performed in support of financial transactions, typically by supervisory personnel. A complete list of transactions supported by the ACI standard POS message is shown in the following tables.

Financial Transactions

The following table describes the financial transactions supported by the ACI standard POS message.

Financial Transactions	
Transaction	**Description**
Additional card activation	Enables customers to purchase an additional stored value card that is associated with an existing stored value card account.
Balance inquiry	Online inquiry into the balance of a customer's account. This transaction can be initiated using a credit or debit card. Both magnetic stripes and ICCs are supported for this transaction.
Card activation	Activates a stored value card account when a customer purchases a stored value card.
Card verification	Online check to see if a customer's credit card is valid. This transaction can be initiated using a credit card. Both magnetic stripes and ICCs are supported for this transaction.

Financial Transactions	
Transaction	**Description**
Cash advance	Cash disbursement from a customer's account at the point of sale. This transaction can be initiated using a credit or debit card. Both magnetic stripes and ICCs are supported for this transaction.
Cash advance adjustment	Corrects an error in a previously completed cash advance transaction. Both magnetic stripes and ICCs are supported for this transaction.
Cash back adjustment	Corrects an error in a previously completed purchase with cash back transaction. Adjustments can be made only to the cash back amount. Both magnetic stripes and ICCs are supported for this transaction.
Check guarantee	Electronically reserves funds in a customer's checking account to cover a check written by the customer. **Note:** When using electronic check authorization, the transaction is authorized and the cardholder account is impacted.
Check verification	Online check verification or authorization to see if a customer's check is tied to a valid checking account.
Full redemption	Enables customers to redeem the remaining value from a stored value card. This transaction reduces the balance of the stored value card account to zero.
Mail or telephone order	Purchase of goods or services where the cardholder is not present at the point of sale (e.g., through the mail or by telephone). This transaction can be initiated using a credit or debit card. Both magnetic stripes and ICCs are supported for this transaction.
Merchandise return	Return of merchandise to a retailer for refund. The refund is electronically credited to the customer account. This transaction can be initiated using a credit or debit card. Both magnetic stripes and ICCs are supported for this transaction.

Financial Transactions	
Transaction	**Description**
Merchandise return adjustment	Corrects an error in a previously completed merchandise return transaction. Both magnetic stripes and ICCs are supported for this transaction.
Normal purchase	Purchase of goods or services at the point of sale using a credit or debit card. Both magnetic stripes and ICCs are supported for this transaction.
Preauthorization purchase	Online check of a customer's account to determine if funds are available for an intended purchase. The amount preauthorized is placed on hold in the customer's account until the purchase is completed, a timer expires, or the hold is removed. This transaction can be initiated using a credit or debit card. Both magnetic stripes and ICCs are supported for this transaction.
Preauthorization purchase completion	Completion of a purchase that was preauthorized. The actual amount of the purchase is entered using this transaction and the hold for the amount entered in the preauthorization purchase transaction is released. The transaction is then completed using the correct amount. This transaction can be initiated using a credit or debit card. Both magnetic stripes and ICCs are supported for this transaction.
Purchase adjustment	Corrects an error in a previously completed purchase transaction. Both magnetic stripes and ICCs are supported for this transaction.
Purchase with cash back	Combination purchase and cash withdrawal at the point of sale. The requested transaction amount is higher than the price of the purchase. The customer receives the difference in cash. This transaction is used only with debit cards. Both magnetic stripes and ICCs are supported for this transaction.
Replenishment	Enables customers to add funds to stored value card. This transaction increases the balance of a stored value card account by the replenishment amount.

Administrative Transactions

The following table describes the administrative transactions supported by the ACI standard POS message.

Administrative Transactions	
Transaction	**Description**
Batch subtotals	Obtains the count and amount of draft capture transactions that have been entered at a POS terminal since the last close batch request transaction or since the totals were last cleared from the terminal data in the host database. These totals include only draft capture transaction activity unless a card type is set up for authorization only processing, in which case the totals contain authorization only totals. However, the host system uses only draft capture totals for terminal reconciliation. Authorization only activity is not included in terminal reconciliation. This transaction can be used when balancing a terminal to determine whether the host system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time and does not affect totals maintained at the host.
Close batch	Ends the current batch and opens a new batch to which data is logged and accumulated for the terminal in the host database. If the terminal can send its own batch totals, the host can also compare the terminal batch totals to the batch totals maintained at the host.

Administrative Transactions	
Transaction	**Description**
Close day	Ends the current terminal day and begins a new day to which data is logged and accumulated for the terminal in the host database. If the terminal can send its own daily totals, the host can also compare the terminal daily totals to the daily totals maintained at the host. **Note:** Close day transactions, when entered, must be immediately preceded by a close shift transaction. If not, the host system denies the close day transaction. Close day transactions should be entered only once during a reporting day in order to maintain reporting integrity.
Close shift	Ends the current shift and opens a new shift to which data is logged and accumulated for the terminal in the host database. If the terminal can send its own shift totals, the host can also compare the terminal shift totals to the shift totals maintained at the host. **Note:** Close shift transactions, when entered, must be immediately preceded by a close batch transaction. If not, the host system denies the close shift transaction. Close shift transactions should be entered only once during a shift in order to maintain reporting integrity.
Day subtotals	Obtains the count and amount of draft capture transactions that have been entered at a point-of-sale terminal since the last close day request transaction or since the totals were cleared from the terminal data in the host database. These totals include only draft capture transaction activity unless a card type is set up for authorization only processing, in which case the totals contain authorization only totals. However, the host system uses only draft capture totals for terminal reconciliation. Authorization only activity is not included in terminal reconciliation.

Administrative Transactions	
Transaction	**Description**
Day totals *continued*	This transaction can be used when balancing a terminal to determine whether the host system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time and does not affect totals maintained at the host.
Download	Causes a full- or a partial-download to be sent to the terminal.
Employee subtotals (Clerk totals)	Obtains the count and amount of transactions that have been entered at a POS terminal by a specific employee since the totals were last cleared from the terminal data in the host database. This includes clerk check amounts and counts. The employee subtotals transaction request also indicates what type of totals to return in the response. The following types of totals can be requested: • All clerk totals for a specific terminal • Totals for a clerk at a specific terminal • Clerk totals for a specific batch for a terminal • Totals for a specific clerk over all terminals This transaction requests clerk totals from the host. It can be used when balancing a terminal to determine whether the host system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time and does not affect totals maintained at the host.
Handshake	Verifies the status of the communications link with the host and, optionally, the terminal's communications and authentication keys.
Logoff	Clears the clerk ID from the terminal data in the host database.

Administrative Transactions	
Transaction	**Description**
Logon	Changes the clerk ID for the terminal data in the host database and accumulates totals for the new employee. It also automatically performs an implied logoff of the previous employee, if required.
Mail delivered	Notifies the host that mail in a read mail request was delivered to the terminal.
Read mail	Obtains a mail message intended for the point-of-sale terminal.
Send mail	Sends a mail message from the point-of-sale terminal to another destination.
Shift subtotals	Obtains the count and amount of draft capture transactions that have been entered at a point-of-sale terminal since the last close shift request transaction or since the totals were last cleared from the terminal data in the host database. These totals include only draft capture transaction activity unless a card type is set up for authorization only processing, in which case the totals contain authorization only totals. However, the host system uses only draft capture totals for terminal reconciliation. Authorization only activity is not included in terminal reconciliation. This transaction can be used when balancing a terminal to determine whether the host system totals are in agreement with the actual totals at the terminal. This transaction can be entered at any time and does not affect totals maintained at the host.

Message Type Support

The ACI standard POS message supports both request and response messages. All request messages are sent from the terminal or controller to the host. All response messages are sent from the host to the terminal or controller. A *message type* and *message subtype* are associated with each request and response message sent. The message type indicates whether the message contains a financial or administrative transaction. The message subtype further identifies the message as being online, store-and-forward, force post, or a reversal. Each of these message subtypes are described below.

Note: Administrative transactions must always be sent online. They cannot be force posted or stored and forwarded.

Online Messages

Online messages are exchanged between a POS device and the host during normal transaction processing when the terminal is connected to the host. A transaction is initiated at the POS device, a request message is sent to the host for authorization, and the host authorizes the transaction and sends a response back to the terminal.

Store-and-Forward Messages

Store-and-forward messages are used when a terminal is offline or otherwise not communicating with the host when the transaction was initiated. The transaction is approved and held in the terminal's memory until communications are resumed, at which time the transaction is sent to the host in a store-and-forward message. The host must accept and post the transaction.

Force Post Messages

Force post messages are messages sent from the POS device to the host for transactions that have been preauthorized. In this case, the host posts the transaction as authorized to the cardholder account. Force post messages are also referred to as advice messages.

Reversal Messages

Reversal messages are sent from a transaction originator to a transaction authorizer indicating that any database impacts from the previous transaction should be reversed because the transaction either could not be completed or was completed incorrectly. The ACI standard POS message supports only whole reversals; it does not support partial reversals.

Reversals can be generated by the terminal or controller because a transaction request times out, a transaction is cancelled by the customer and reversed by a clerk at the terminal, a message authentication code (MAC) could not be authenticated, or a transaction could not be properly completed (e.g., a controller is unable to send a response to the individual terminal that generated the transaction).

2: The ACI Standard POS Message

This section contains the following information concerning the ACI standard POS message:

- Standard message header
- Optional data fields
- Request message requirements
- Response message requirements

Every message sent between the terminal and the host contains a standard message header. The header contains information about the terminal, the transaction, and the employee who initiated the transaction.

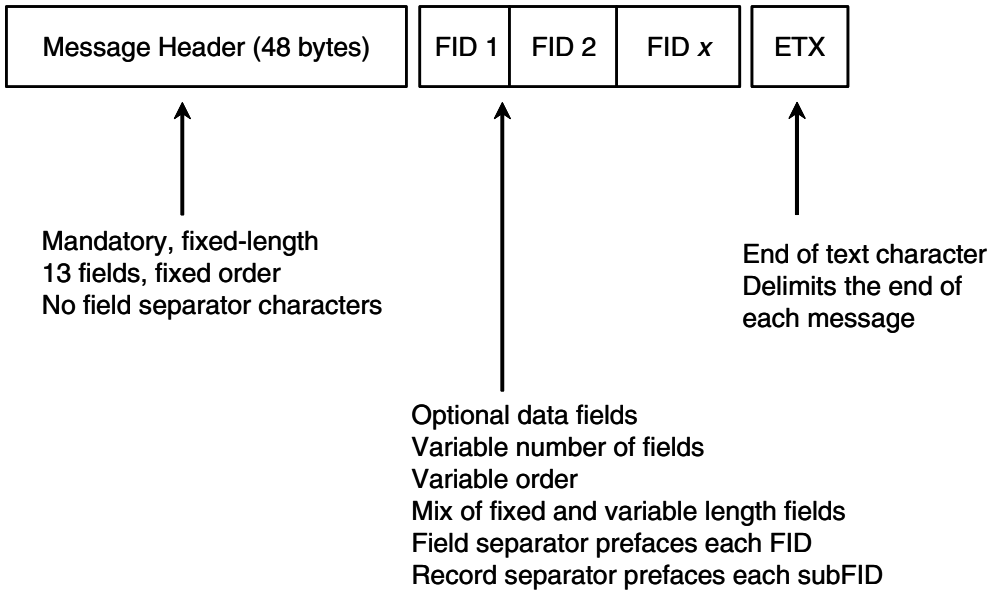
Although the standard header can stand alone, any number of optional data fields can follow it. The only restrictions on the number of optional data fields that can be included in the message are the protocol limitations, the size of the terminal read buffer, and the size of the host read buffer. The host read buffer is configurable to a maximum of 4,088 bytes, including all data communication control characters.

Note: All data carried in an ACI standard POS message must be in ASCII format. Any binary coded data used at the POS device must be converted into ASCII before being included in a request message to the host. This conversion must be reversed to retrieve binary coded data from an ASCII response received from the host.

Each optional data field included in the message is identified with a field identifier (FID) character preceding the optional data field. In addition, each FID is prefaced with a field separator character—a period (.), which is represented by the hexadecimal characters 1C. Some optional data fields contain subfields that are identified by a subfield identifier (SFID) character. Each SFID is separated with a record separator character—an exclamation point (!), which is represented by the hexadecimal characters 1E.

Each request and response message must be surrounded by the appropriate protocol control characters. The end of each message is always delimited by an end-of-text (ETX) character.

The following diagram summarizes the major components of the ACI standard POS message, excluding most data control characters.



Encrypted Message Format

The ACI standard POS message supports full message encryption and configurable message encryption. Full message encryption allows the institution to encrypt all optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. Configurable message encryption allows the institution to encrypt specific optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. The optional data fields are configured to be encrypted in the ACI standard device configuration file request and response field maps. Full message encryption and configurable message encryption are enabled using the setting in the POS terminal data file.

A distinct separator character is defined to delimit the encrypted portion of an ACI standard POS message as follows: Hexadecimal 1F (Octal 37).

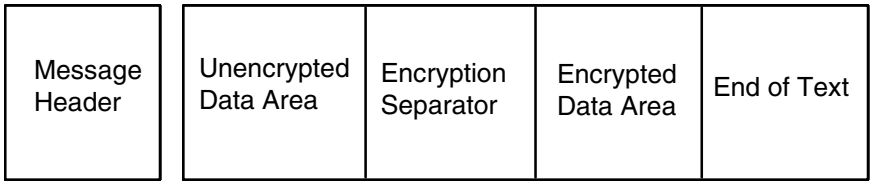
The encryption separator marks the beginning of encrypted data in the message. All data between the encryption separator and the end of the message (not including the end-of-text character) is considered to be encrypted under the data encryption key. The following rules must be observed when encrypting data in the ACI standard POS message:

- The standard message header must not be encrypted. Only optional data fields and subfields can be encrypted.
- All unencrypted optional data fields and subfields must immediately follow the standard message header.
- The encryption separator must immediately follow the last unencrypted optional data field or subfield.
- All encrypted optional data fields and subfields must immediately follow the encryption separator.
- Field separator characters (hexadecimal 1C) and field identifiers (FIDs), which preface optional data fields, are included in the encrypted data.
- Record separator characters (hexadecimal 1E) and subfield identifiers (subFIDs), which preface optional data subfields, are included in the encrypted data.
- If an end-of-text (ETX) character is appended to the message, it must not be encrypted.

Whether using full message encryption or configurable message encryption, the format of the ACI Standard POS Message is the same. The message begins with the standard message header, which is never encrypted. Following the standard message header is the unencrypted data area, which contains all the optional data

fields and subfields that are not encrypted under the data encryption key. Next comes the encryption separator (hexadecimal 1F), a separator character that is used to mark the beginning of the encrypted data area. The encrypted data area extends from the encryption separator to the end of the message and contains all of the optional data fields and subfields that are encrypted under the data encryption key.

Field separator characters (hexadecimal 1C) and field identifiers (FIDs), which preface optional data fields, are included in the encrypted data. Likewise, record separator characters (hexadecimal 1E) and subfield identifiers (subFIDs), which preface optional data subfields, are included. The encryption separator itself is not encrypted, nor is the end-of-text character at the end of the message, if present. The following diagram is an illustration of the format of an encrypted ACI Standard POS message.



By grouping optional data fields and subfields into the encrypted data area, the SPDH is able to encrypt or decrypt the data with a single call to the host security module (HSM).

Binary Data Conversion

All data in an ACI standard POS message must be in ASCII format. Binary data used at the POS device must be converted into ASCII format before being included in a request message to the host. This conversion must be reversed to retrieve the binary data from an ASCII response received from the host. Hexadecimal characters are used to represent binary data in ASCII formatted messages as described below.

Several fields in EMV request and response data contain binary data. For conversion, the binary data is divided into groups of four bits. Each group of four bits is assigned a hexadecimal (hex) character. Thus, eight bits of binary data is represented by two hexadecimal characters. It is the hexadecimal characters that are carried in the ACI standard POS device message.

Each possible combination of the four bit values is assigned a hexadecimal character value according to the following table.

Conversion Table			
Bit Value	**Hex Value**	**Bit Value**	**Hex Value**
0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

In all conversions, bit 0 of the binary data is always considered to be the most significant (i.e., “leftmost”) bit. Thus, for eight bits of binary data in a field, bits 0–3 are represented by the first hexadecimal character in the field, while bits 4–7 are represented by the second hexadecimal character in the field.

Note: In EMV specifications, bits are numbered from 8 to 1, with bit 8 considered to be the most significant bit. Thus, bit 8 in EMV specifications is represented by bit 0 in this manual, while bit 1 in EMV specifications is represent by bit 7 in this manual. The following table shows the EMV bit numbering scheme for the bits carried in the ACI standard POS message. The ACI bit numbering scheme is used for the applicable fields described in this manual.

Bit Numbering			
ACI	**EMV**	**ACI**	**EMV**
0	8	4	4
1	7	5	3
2	6	6	2
3	5	7	1

The following example illustrates how eight bits of binary data are converted by a POS device for placement in the Cryptographic Information Data field of an EMV request message sent to the host.

Bits	Binary Data	Hex Values
0–3	1000	8
4–7	1011	B

Bits 0–3 of binary data are represented by the first hexadecimal character, while bits 4–7 are represented by the second hexadecimal character. In this example message, the cryptogram is an authorization request cryptogram (ARQC), issuer authentication failed, and an advice is required. The binary data for this information is 10001011, which is represented as “8B” in the ACI standard POS device message.

In EMV request messages, the EMV Request Data (FID 6, subFID O) field must contain the hexadecimal equivalents of binary data. In EMV response messages, the following fields must contain the hexadecimal equivalents of binary data.

- EMV Response Data (FID 6, subFID Q)
- EMV Reversal Data/EMV Additional Response Data (FID 6, subFID R)

Standard Message Header

The standard header must be in every message sent from the terminal to the host and will be in every response sent from the host to the terminal. It consists of 13 fields, totaling 48 bytes, and is not configurable. The standard message header identifies information such as the type of message being sent, the employee entering the transaction, the ID of the terminal where the transaction was entered, and the transaction code associated with the transaction.

The customer and the terminal vendor decide the manner in which values in the standard header are determined. For example, when the values in the processing flag fields need to be changed, the customer and the vendor must agree on the manner in which to change the values.

Standard message header fields are summarized in the table below. Following the table are individual descriptions of the fields. Field descriptions include possible values for each of these fields. Fields not used in requests must be blank-filled or zero-filled. A value of a blank space is indicated with the symbol ␣.

Standard Message Header Structure

The following table describes the structure of the standard message header—including for each of its fields, the positions occupied by the field, the length and format of the field, and the name of the field.

Position	Field Length	Field Name
1–2	2 alphanumeric characters	Device Type
3–4	2 numeric characters	Transmission Number
5–20	16 alphanumeric characters	Terminal ID
21–26	6 alphanumeric characters	Employee ID
27–32	6 numeric characters	Current Date
33–38	6 numeric characters	Current Time
39	1 alphanumeric character	Message Type
40	1 alphanumeric character	Message Subtype
41–42	2 numeric characters	Transaction Code

Position	Field Length	Field Name
43	1 numeric character	Processing Flag 1
44	1 numeric character	Processing Flag 2
45	1 numeric character	Processing Flag 3
46–48	3 numeric characters	Response Code

Positions 1–2 — Device Type

A code that can be used by controllers on nonswitched lines to identify individual terminals. This field can also be used by the host to route messages from dial-up terminals to the appropriate application process. Valid values are as follows:

9. = Dial or leased line terminal or network
Other = As defined

Positions 3–4 — Transmission Number

A two-digit number used by the host to detect and drop duplicate requests. If this field contains a nonzero value, and it is equal to the transmission number of the last request and the previous response was an approval, the host drops the request and sends a response indicating that it received a duplicate request and that it was dropped. This field is right-justified and zero filled. Valid values are as follows:

00 = Transmission number not checked
Other = Transmission number checked for duplicates

Positions 5–20 — Terminal ID

A value that uniquely identifies the terminal. It is used by the host as the key to the appropriate record for the terminal in the host database. This value must be manually entered at the terminal initially and whenever the value has been corrupted. This field is left-justified and blank filled.

Positions 21–26 — Employee ID

One to six alphanumeric characters that uniquely identify the employee entering the transaction. It is used by the host to maintain employee totals. This field is left-justified and blank filled. For request headers, the valid values are as follows:

␣␣␣␣␣␣ = Employee ID is not validated (where ␣ is a space)
Other = Employee ID validated against the terminal data in the host database

For response headers, this information is echoed from the request.

Positions 27–32 — Current Date

The date (YYMMDD) of the transaction. For requests, this field is optional and contains the local terminal date. If it is not provided in the request, or is invalid or incorrect, the host defaults to the current system date. For responses, the host returns the current system date taking into account differing time zones. The value in this field must be echoed in terminal-generated reversals.

If this field contains a value in the request, the Current Time field also must contain a valid value. The device must send both fields if the current system date and time of the host are not to be used.

In message authentication code (MAC) reversal messages from the terminal, the value in this field must match the date on the response being reversed.

Positions 33–38 — Current Time

The time (hhmmss) of the transaction, where 000000 is midnight. For requests, this field is optional and contains the local terminal time. If it is not provided in the request, or is invalid or incorrect, the host defaults to the current system time. For responses, the host returns the current system time taking into account differing time zones. The value in this field must be echoed in terminal-generated reversals.

If this field contains a value in the request, the Current Date field also must contain a valid value. The device must send both fields if the current system date and time of the host are not to be used.

In message authentication code (MAC) reversal messages from the terminal, the value in this field must match the time on the response being reversed.

Position 39 — Message Type

A code that identifies whether the transaction is financial or administrative. Together with the value in the Transaction Code field, this field identifies the type of transaction to the host. A listing of the various combinations is included later in this section. Valid values are as follows:

A = Administrative transaction
F = Financial transaction
L = Pass-through administrative transaction (formerly called merchant link)
M = Pass-through financial transaction (formerly called merchant link)

Position 40 — Message Subtype

A code that identifies the message as being either online, store-and-forward, force-post, or a reversal. Reversals are generated by the terminal or controller, or because of a message authentication code (MAC) problem. Valid values are as follows. All valid values are uppercase letters.

A = Timeout reversal—advice. Placed in this field by the terminal or controller when a store-and-forward transaction must be reversed because of a timeout.

C = Terminal or controller reversal. Placed in this field by the terminal or the controller when the transaction must be reversed. The original response message is then sent back to the host, with the exception of this field, thus reversing the transaction. A controller might place a C in this field if the controller is unable to send the response to the individual terminal that generated the transaction.

D = Terminal decryption reversal. Produced when a balance inquiry transaction must be reversed due to a data decryption error by the POS/PIN pad device. The response message is then sent back to the host as received with this field set to a value of D to reverse the transaction. This value is used only with balance inquiry transactions when the available balance cannot be decrypted.

E = Europay, MasterCard, and Visa (EMV) chip card log-only. Indicates that the transaction was approved by the terminal or EMV card offline. When an EMV transaction is authorized offline by the terminal/EMV card, the card generates a Transaction Certificate (TC). The TC provides information about the transaction which can be used if the transaction is disputed. The TC is found in the Application Cryptogram (AC) field within the FID 6 subFID O (EMV Request Data). The terminal uploads EMV log-only transactions one at a time to the host.

F = Force-post. Indicates the transaction is to be force-posted. Force posting is the manual posting of a transaction to an account. For example, if a POS device becomes inoperable but a merchant wants a transaction to be posted for settlement purposes, the transaction can be posted manually to the account.

O = Online. Indicates the terminal or controller is online to the host. Note: Administrative transaction messages must have a subtype value of O.

R = MAC reversal. Placed in this field by the terminal or controller when it receives a response message containing a MAC that does not match the MAC computed by the terminal. The original response message is then sent back to the host as received, with the exception of this field, thus reversing the transaction.

S = Store-and-forward. Indicates the terminal or controller was offline or otherwise not communicating with the host when the transaction was initiated. The transaction was approved and held in the terminal memory until communications resumed, at which time the transaction was sent to the host. The host must accept and post the transaction.

T = Timeout reversal—online. Placed in this field by the terminal or controller when an online transaction must be reversed because of a timeout.

U = Customer-canceled reversal. Placed in this field by the terminal when a transaction is reversed by a clerk at the terminal.

V = EMV log-only cancellation. Placed in this field by the terminal or the controller when an EMV log-only transaction (subtype E) needs to be cancelled.

Positions 41–42 — Transaction Code

A code that identifies the transaction type associated with the message. Together with the value in the Message Type field, this code identifies the type of transaction sent to the host. A listing of the various combinations is included later in this section. Valid values are as follows:

00 = Normal purchase
01 = Preauthorization purchase
02 = Preauthorization purchase completion
03 = Mail or telephone order
04 = Merchandise return
05 = Cash advance
06 = Card verification
07 = Balance inquiry
08 = Purchase with cash back
09 = Check verification
10 = Check guarantee
11 = Purchase adjustment
12 = Merchandise return adjustment
13 = Cash advance adjustment
14 = Cash back adjustment
15 = Card activation
16 = Additional card activation
17 = Replenishment
18 = Full redemption
50 = Logon request
51 = Logoff request
60 = Close batch request
61 = Close shift request
62 = Close day request
64 = Employee subtotals request
65 = Batch subtotals request
66 = Shift subtotals request
67 = Day subtotals request
70 = Read mail request
71 = Mail delivered request
75 = Send mail request
90 = Download request
95 = Handshake request
96 = Key change request

Position 43 — Processing Flag 1

A code that is used in requests to direct the host to disconnect after its response, or to remain connected for more transactions.

If this field is set to not disconnect the terminal, the host maintains the line indefinitely. In dial-up environments, it is the responsibility of the terminal to initiate a disconnect, either by sending a request (e.g., a Handshake request) with this field set to disconnect, or by dropping the line. If the terminal disconnects by dropping the line, this can be treated as an error condition by the protocol.

In general, when using Visa or POS-2 protocols, the terminal cannot determine whether subsequent requests are pending and should set this field to disconnect. When the protocol receives a message from the terminal, the protocol determines whether more requests exist for the host in its queue before the host issues a disconnect message to the terminal.

Furthermore, the terminal cannot determine if subsequent requests are pending during a download if the terminal cannot determine whether the host response indicates the presence or absence of more download data. The host sends a disconnect message to the terminal if the host determines there is no more download data.

An example of this situation is a full download that requires multiple request and response messages. For downloads, the host sends an 880 response code indicating that no more data exists or an 881 response code indicating that more data is forthcoming. Thus, this field should be set to disconnect for downloads. Then, when there is no more data to be sent, the host issues a disconnect message to the terminal. For requests, the valid values are as follows:

0 = Respond and disconnect
1 = Respond and do not disconnect

This field is used in responses to inform the terminal of waiting mail. For responses, the valid values are as follows:

0 = No mail waiting
1 = Mail waiting

Position 44 — Processing Flag 2

The use of this code depends on the value in the Message Type field.

For pass-through administrative transactions (message type L), this code is used in requests to indicate whether concurrent processing can be used. The valid values are as follows:

0 = No, transactions cannot be processed concurrently.
1 = Yes, transactions can be processed concurrently.

If concurrent processing is used, the PIN Pad ID will be capable of processing multiple credit card transactions at the same time.

For all other transactions, this code is used in requests to indicate whether the POS device is EMV capable. The valid values are as follows:

0 = No, the terminal is not EMV capable.
5 = Yes, the terminal is EMV capable.

For all transactions, this code is used in responses to inform the terminal that it should request a download. For responses, the valid values are as follows:

0 = No download waiting
1 = Download waiting

Position 45 — Processing Flag 3

The use of this code depends on the value in the Message Type field.

For failed pass-through administrative transactions (message type L), this code is used in requests to indicate whether concurrent processing can be used. If concurrent processing is used, the PIN Pad ID will be capable of processing multiple credit card transactions at the same time. The valid values are as follows:

0 = No, transactions cannot be processed concurrently.
1 = Yes, transactions can be processed concurrently.

For all other requests, this code is used to indicate the totals to return in response to a clerk totals request. This field is not used for responses. Valid values are as follows:

0 = All clerk totals for a specific terminal
1 = Totals for a clerk at a specific terminal
2 = Clerk totals for a specific batch for a terminal
3 = Totals for a specific clerk over all terminals

Positions 46–48 — Response Code

This code is not used in requests.

This code is used in responses to inform the terminal of the transaction processing results. Valid values are as follows:

Approved Codes
000 = Approved balances available
001 = Approved no balances available
002 = Approved country club status
003 = Approved (maybe more identification is required)
004 = Approved pending identification (sign paper draft is required)
005 = Approved blind
006 = Approved VIP status
007 = Approved administrative transaction
008 = Approved negative card file hit OK
009 = Approved commercial status
010 = Approved for a partial amount

Declined Codes
050 = General
051 = Expired card
052 = Number of PIN tries exceeded
053 = No sharing allowed
054 = No security module
055 = Invalid transaction
056 = Transaction not supported by institution
057 = Lost or stolen card
058 = Invalid card status
059 = Restricted status
060 = Account not found in cardholder database
061 = Positive balance account record not found
062 = Positive balance account update error

063 = Invalid authorization type in institution database
064 = Bad track information
065 = Adjustment not allowed in institution database
066 = Invalid credit card advance increment
067 = Invalid transaction date
068 = Transaction log file error
069 = Bad message edit
070 = No institution database record
071 = Invalid routing to host application
072 = Card on national negative file
073 = Invalid routing authorization service
074 = Unable to authorize
075 = Invalid PAN length
076 = Insufficient funds in positive balance account
077 = Preauthorization full
078 = Duplicate transaction received
079 = Maximum online refund reached
080 = Maximum offline refund reached
081 = Maximum credit per refund reached
082 = Maximum number of times used
083 = Maximum refund credit reached
084 = Customer selected negative card file reason
085 = Inquiry not allowed—no balances
086 = Over floor limit
087 = Maximum number refund credits reached
088 = Place call
089 = Card status equals 0 (inactive) or 9 (closed)
090 = Referral file full
091 = Problem accessing negative card file
092 = Advance less than minimum
093 = Delinquent
094 = Over limit table or exceeds amount available
095 = Amount over maximum
096 = PIN required
097 = Mod 10 check
098 = Force post
099 = Could not access positive balance account in database

Referral Codes

100 = Unable to process transaction
101 = Unable to authorize—issue call
102 = Call
103 = Problem accessing negative card file
104 = Problem accessing cardholder account

105 = Card not supported
106 = Amount over maximum
107 = Over daily limit
108 = Card authorization parameters not found
109 = Advance less than minimum
110 = Number times used
111 = Delinquent
112 = Over limit table
113 = Timeout
115 = Transaction log file full
120 = Problem accessing cardholder usage accumulation data
121 = Problem accessing administrative card data
122 = Unable to validate PIN; security module is down
130 = Authorization request cryptogram (ARQC) referral
131 = Card verification results (CVR) referral
132 = Terminal verification results (TVR) referral
133 = Reason online code referral
134 = Fallback referral

Service Code

150 = Merchant not on file

Transaction Error Codes

200 = Invalid account
201 = Incorrect PIN
202 = Cash advance is less than minimum
203 = Administrative card needed
204 = Enter lesser amount. Note: Response code 204 has two meanings. This code can also be used when the transaction amount exceeds the retailer ceiling limits.
205 = Invalid advance amount
206 = Cardholder authorization data not found
207 = Invalid transaction date
208 = Invalid expiration date
209 = Invalid transaction code
251 = Cash back exceeds daily limit
400 = Authorization request cryptogram (ARQC) failure
401 = Hardware security module parameter error
402 = Hardware security module failure
403 = Integrated circuit card key information not found
404 = Application transaction counter (ATC) check failure
405 = Card verification results (CVR) decline
406 = Terminal verification results (TVR) decline
407 = Reason online code decline

408 = Fallback decline
800 = Format error
801 = Invalid data
802 = Invalid employee number
809 = Invalid close transaction
810 = Transaction timeout
811 = System error
820 = Invalid terminal identifier
821 = Invalid response length

Mail and Download Codes

870 = Mail delivered
871 = Mail stored
880 = Mail message has been received in its entirety
881 = Mail message received successfully and there is more data for this mail message
880 = Download has been received in its entirety
881 = Download received successfully and there is more data for this download
882 = Download aborted (call for service)

Decline Codes

878 = Incorrect PIN length error
889 = MAC communications key (KMAC) synchronization error
898 = Invalid MAC
899 = Sequence error—resync

POS Capture Codes

900 = Number of PIN tries exceeded
901 = Expired card
902 = Negative card file capture code
903 = Card status is 3 (stolen)
904 = Advance less than minimum
905 = Number times used exceeded
906 = Delinquent
907 = Over limit table
908 = Amount over maximum
909 = Capture
910 = Authorization request cryptogram (ARQC) capture
911 = Card verification results (CVR) capture
912 = Terminal verification results (TVR) capture

Decline Administrative Card—Call Required Codes

950 = Administrative card not found
951 = Administrative card not allowed
959 = Administrative transactions not supported

Approved Administrative Codes

952 = Approved administrative request—in window
953 = Approved administrative request—out of window
954 = Approved administrative request—anytime

Chargeback Codes

955 = Chargeback—customer file updated
956 = Chargeback—customer file updated—acquirer not found
957 = Chargeback—incorrect prefix number
958 = Chargeback—incorrect response code or card prefix configuration
960 = Chargeback—approved customer file not updated
961 = Chargeback—approved customer file not updated, acquirer not found
962 = Chargeback—accepted, incorrect destination

Standard Message Header Examples

This subsection provides examples of the standard message headers used for normal purchase, merchandise return, and mail or telephone order transactions. These examples are designed to show how the standard message header is formatted in different transaction scenarios. Each example contains three parts. The first part explains the transaction scenario being described. The second part illustrates how the standard message header is formatted in the request from the terminal to the host. The third part illustrates how the standard message header is formatted in a response from the host to the terminal.

In these examples, the symbol ♭ represents a space, and quotation marks (“ ”) are used to delimit the text and are not part of the header.

Normal Purchase Transaction

The examples shown below describe the standard message headers used in a typical request and response sequence for a normal purchase transaction.

Request

The following table describes the fields in the standard message header in a sample request for a normal purchase transaction.

Field	Description
Device Type	9. (dial or leased line terminal, link type: switched)
Transmission Number	00 (transmission number not checked)
Terminal ID	POS-TERMINAL-001
Employee ID	♭♭♭♭♭♭ (employee ID not validated)
Current Date	000000 (current date is not applicable in requests)
Current Time	000000 (current time is not applicable in requests)
Message Type	F (normal purchases are financial transactions)

Field	Description
Message Subtype	O (online)
Transaction Code	00 (normal purchase)
Processing Flag 1	0 (respond and disconnect)
Processing Flag 2	0 (terminal is not EMV-capable)
Processing Flag 3	0 (all clerk totals for a specific terminal)
Response Code	000 (response code is not applicable in requests)

Using the values set in the table above, the standard message header for this request would be formatted as follows, where ␣ denotes a blank space:

```
Request "9.00POS-TERMINAL-001␣␣␣␣␣␣000000000000FO00000000"
```

Response

The following table describes the fields in the standard message header in a sample response to a normal purchase transaction request.

Field	Description
Device Type	9. (echoed from request)
Transmission Number	00 (echoed from request)
Terminal ID	POS-TERMINAL-001 (echoed from request)
Employee ID	␣␣␣␣␣␣ (echoed from request)
Current Date	010710 (local terminal date YYMMDD)
Current Time	121005 (local terminal time hhmmss)
Message Type	F (echoed from request)
Message Subtype	O (echoed from request)
Transaction Code	00 (echoed from request)

Field	Description
Processing Flag 1	0 (no mail waiting)
Processing Flag 2	0 (no download waiting)
Processing Flag 3	0 (processing flag 3 is not applicable in responses)
Response Code	001 (approved with no balances)

Using the values set in the table above, the standard message header for this response would be formatted as follows, where ␣ denotes a blank space:

```
Response "9.00POS-TERMINAL-001␣␣␣␣␣␣␣010710121005FO00000001"
```

Merchandise Return Transaction

The following examples describe the standard message headers used in a typical request and response sequence for a merchandise return transaction. The standard message header field settings are also explained for each example.

Request

The following table describes the fields in the standard message header in a sample request for a merchandise return transaction.

Field	Description
Device Type	9. (dial or leased line terminal, link type: switched)
Transmission Number	23 (transmission number checked for duplicates)
Terminal ID	POS-TERMINAL-056
Employee ID	987654 (employee ID is validated)
Current Date	000000 (current date is not applicable in requests)

Field	Description
Current Time	000000 (current time is not applicable in requests)
Message Type	F (merchandise returns are financial transactions)
Message Subtype	O (online)
Transaction Code	04 (merchandise return)
Processing Flag 1	1 (respond and do not disconnect)
Processing Flag 2	0 (terminal is not EMV-capable)
Processing Flag 3	0 (all clerk totals for a specific terminal)
Response Code	000 (response code is not applicable in requests)

Using the values set in the table above, the standard message header for this request would be formatted as follows, where ♭ denotes a blank space:

```
Request "9.23POS-TERMINAL-056987654000000000000FO04100000"
```

Response

The following table describes the fields in the standard message header in a sample response to a merchandise return transaction request.

Field	Description
Device Type	9. (echoed from request)
Transmission Number	23 (echoed from request)
Terminal ID	POS-TERMINAL-056 (echoed from request)
Employee ID	987654 (echoed from request)
Current Date	010710 (local terminal date YYMMDD)
Current Time	192122 (local terminal time hhmmss)

Field	Description
Message Type	F (echoed from request)
Message Subtype	O (echoed from request)
Transaction Code	04 (echoed from request)
Processing Flag 1	0 (no mail waiting)
Processing Flag 2	0 (no download waiting)
Processing Flag 3	0 (processing flag 3 is not applicable in responses)
Response Code	057 (declined with lost or stolen card status)

Using the values set in the table above, the standard message header for this response would be formatted as follows, where ♭ denotes a blank space:

```
Response "9.23POS-TERMINAL-0569876540107101921220FO04000057"
```

Mail or Telephone Order

The examples shown below describe the standard message headers used in a typical request and response sequence for a mail or telephone order transaction.

Request

The following table describes the fields in the standard message header in a sample request for a mail or telephone order transaction.

Field	Description
Device Type	bb (controller on non-switched line to identify individual terminals, link type: non-switched)
Transmission Number	54 (transmission number checked for duplicates)
Terminal ID	POS-TERMINAL-589
Employee ID	123456 (employee ID is validated)

Field	Description
Current Date	000000 (current date is not applicable in requests)
Current Time	000000 (current time is not applicable in requests)
Message Type	F (mail or telephone orders are financial transactions)
Message Subtype	S (store-and-forward)
Transaction Code	03 (mail or telephone order)
Processing Flag 1	0 (respond and disconnect)
Processing Flag 2	0 (terminal is not EMV-capable)
Processing Flag 3	0 (all clerk totals for a specific terminal)
Response Code	000 (response code is not applicable in requests)

Using the values set in the table above, the standard message header for this request would be formatted as follows, where ♭ denotes a blank space:

```
Request "bb54POS-TERMINAL-589123456000000000000FS03000000"
```

Response

The following table describes the fields in the standard message header in a sample response to a mail or telephone order transaction request.

Field	Description
Device Type	bb (echoed from request)
Transmission Number	54 (echoed from request)
Terminal ID	POS-TERMINAL-589 (echoed from request)
Employee ID	123456 (echoed from request)
Current Date	010710 (local terminal date YYMMDD)

Field	Description
Current Time	192122 (local terminal time hhmmss)
Message Type	F (echoed from request)
Message Subtype	S (echoed from request)
Transaction Code	03 (echoed from request)
Processing Flag 1	0 (no mail waiting)
Processing Flag 2	0 (no download waiting)
Processing Flag 3	0 (processing flag 3 is not applicable in responses)
Response Code	001 (approved with no balances)

Using the values set in the table above, the standard message header for this response would be formatted as follows, where ␣ denotes a blank space:

```
Response "bb54POS-TERMINAL-589123456010710192122FS03000001"
```

Optional Data Fields

After the standard message header, the remaining portion of the message consists of a series of optional data fields. Optional data fields can be included in requests from the terminal and responses from the host for each transaction type and are identified in the system by Field Identifiers (FIDs).

Optional data fields are summarized in a table below, followed by individual descriptions of the fields.

Optional data fields 6 through 9 contain subfields that supply additional optional information. These subfields are identified by Subfield Identifiers (SFIDs). For more information about these subfields, see the "Optional Data Subfield" topics later in this section.

Summary Table

The optional data fields are summarized in the table below in order by FIDs, capital letters first, followed by lowercase letters, followed by numbers. The table lists the FID, its picture clause, the length of the field in the message, and its associated field name. In addition, a check mark (✓) appears in the RQST or RESP columns if the optional data field is available for requests or responses, respectively.

FID	Picture	Length	Field Name	RQST	RESP
A	PIC X(20)	1 to 20 bytes	Customer Billing Address	✓	✓
B	PIC 9(18)	1 to 18 bytes	Amount 1	✓	✓
C	PIC 9(18)	1 to 18 bytes	Amount 2	✓	✓
D	PIC 9(1)	1 byte	Application Account Type	✓	✓
E	PIC 9(19)	1 to 19 bytes	Application Account Number		✓
F	PIC X(8)	8 bytes	Approval Code	✓	✓
G	PIC X(8)	8 bytes	Authentication Code	✓	✓
H	PIC X(48)	16 to 48 bytes	Authentication Key		✓
I	PIC X(48)	16 to 48 bytes	Data Encryption Key		✓

FID	Picture	Length	Field Name	RQST	RESP
J	PIC 9(18)	18 bytes	Available Balance		✓
K	PIC 9(6)	6 bytes	Business Date	✓	✓
L	PIC 9(1)	1 byte	Check Type	✓	✓
M	PIC X(48)	16 to 48 bytes	Communications Key		✓
N	PIC X(40)	1 to 40 bytes	Customer ID	✓	✓
O	PIC 9(2)	2 bytes	Customer ID Type	✓	✓
P	PIC 9(1)	1 byte	Draft Capture Flag	✓	✓
Q	PIC X(16)	1 to 16 bytes	Echo Data	✓	✓
R	PIC X(1)	1 byte	Card Type	✓	✓
S	PIC X(10)	1 to 10 bytes	Invoice Number	✓	✓
T	PIC X(10)	1 to 10 bytes	Invoice Number/Original	✓	✓
U	PIC X(1)	1 byte	Language Code	✓	✓
V	PIC X(15)	15 bytes	Mail/Download Key	✓	✓
W	PIC X(449) for requests PIC X(957) for responses	1 to 957 bytes	Mail Text/Download Data	✓	✓
X	PIC 9(3)	3 bytes	ISO Response Code		✓
Y	PIC X(9)	1 to 9 bytes	Customer ZIP Code	✓	✓
Z	PIC X(1)	1 byte	Address Verification Status Code		✓
a	PIC X(250)	1 to 250 bytes	Optional Data	✓	✓
b	PIC X(16)	16 bytes	PIN/Customer	✓	
c	PIC X(16)	16 bytes	PIN/Supervisor	✓	
d	PIC 9(12)	1 to 12 bytes	Retailer ID	✓	✓

FID	Picture	Length	Field Name	RQST	RESP
e	PIC 9(2)	2 bytes	POS Condition Code	✓	✓
f	PIC 9(2) for requests PIC X(200) for responses	1 to 200 bytes	PIN Length or Receipt Data	✓	✓
g	PIC X(48)	1 to 48 bytes	Response Display		✓
h	PIC X(10)	10 bytes	Sequence Number	✓	✓
i	PIC X(9)	9 bytes	Sequence Number/Original	✓	✓
j	PIC X(2)	2 bytes	State Code	✓	
k	PIC 9(6) for requests PIC X(25) for responses	0 to 25 bytes	Birth Date/Drivers License Expiration Date (for requests)/Terminal Location (for responses)	✓	✓
l	PIC X(75)	75 bytes	Totals/Batch	✓	✓
m	PIC X(75)	75 bytes	Totals/Day	✓	✓
n	PIC X(75)	75 bytes	Totals/Employee		✓
o	PIC X(75)	75 bytes	Totals/Shift	✓	✓
q	PIC X(40)	1 to 40 bytes	Track 2/Customer	✓	✓
r	PIC X(40)	1 to 40 bytes	Track 2/Supervisor	✓	✓
s	PIC X(24)	1 to 24 bytes	Transaction Description		✓
t	PIC X(16)	16 bytes	PIN Pad Identifier	✓	✓
u	PIC X(6)	6 bytes	Acceptor Posting Date		✓
0	PIC X(118)	46 to 118 bytes	American Express Data Collection	✓	✓
1	PIC X(24)	24 bytes	PS2000 Data	✓	✓
2	PIC X(82)	1 to 82 bytes	Track 1/Customer	✓	✓

FID	Picture	Length	Field Name	RQST	RESP
3	PIC X(82)	1 to 82 bytes	Track 1/Supervisor	✓	✓
4	PIC X(171)	156 to 171 bytes	Industry Data	✓	✓
6	variable	variable	Product SubFIDs	✓	✓
7	variable	variable	Product SubFIDs	✓	✓
8	variable	variable	Product SubFIDs	✓	✓
9	variable	variable	Customer SubFIDs	✓	✓

FID A — Billing Address

Request: Optional. Variable length of 1 to 20 bytes.

Response: Optional. Fixed length of 20 bytes. If this field is included in the request message, then it can be echoed in the response.

The Billing Address is the cardholder's billing address. This address is used for address verification.

FID B — Amount 1

Request: Required for the following transactions. Variable length of 1 to 18 bytes.

Normal purchase
Preauthorization purchase
Preauthorization purchase completion
Mail or telephone order
Merchandise return
Cash advance
Purchase with cash back
Check guarantee
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment

Response: Optional. Fixed length of 18 bytes. If included in the request, the value can be echoed in the response. This field is right-justified and zero filled.

Amount 1 is the primary amount field. For transactions that involve one amount, this is the transaction amount. For transactions that involve more than one amount, this is the original or total amount.

For an EMV request, the data in this field is used for the AMT-AUTH field of the EMV Request Data token.

For a transaction that is authorized for a lesser or a greater amount than requested, this field contains the authorized amount in the response message.

FID C — Amount 2

Request: Required for the following transactions. Variable length of 1 to 18 bytes.

Purchase with cash back
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment

Response: Optional. Fixed length of 18 bytes. If included in the request, it can be echoed in the response. This field is right-justified and zero filled.

Amount 2 is the secondary amount field.

For transactions with two amounts involved, Amount 2 is the revised amount.

For transactions with cash back, Amount 2 is the cash back amount.

For an EMV request, this field is used for the AMT-OTHER field of the EMV Request Data token.

For a passthrough financial transaction (message type M) that is a preauthorization with cash back or a preauthorization completion with cash back, this field contains the original transaction amount.

For a purchase with cash back or a preauthorization purchase with cash back transaction that is authorized for a lesser amount, this field contains the reduced cash back amount.

FID D — Application Account Type

Request: Optional. Fixed length of 1 byte.

Response: Optional. Fixed length of 1 byte. If this field is included in the request message, it can be echoed in the response. If this field is not in the request message, it reflects the default account type in a host database.

The Application Account Type indicates the type of account for the transaction. If not submitted by the terminal, the host will use a default account type. Valid values are as follows:

0 = None (use default account type on host)
1 = Checking
2 = Savings
4 = Credit
5 = iDebit
8 = Food stamps
9 = Cash benefit

FID E — Application Account Number

Request: Not available.

Response: Optional. Variable length of 1 to 19 bytes. Trailing blanks are removed. This field is left-justified and blank filled.

The Application Account Number indicates the account against which the transaction was processed. This field originates from the authorizer.

FID F — Approval Code

Request: Optional. Fixed length of 8 bytes.

Response: Optional. Fixed length of 8 bytes. This field is left-justified and blank filled. If this field is included in the request message, then it can be echoed in the response. If this field is not included in the request message, it can be generated by the host application, another host, or an interchange.

The Approval Code represents a unique number generated by the authorizer for the transaction.

FID G — Authentication Code

Request: Optional. Fixed length of 8 bytes. If included, the MAC is verified.

Response: Optional. If responses are verified using MACs, this field is required. Fixed length of 8 bytes.

The Authentication Code contains the eight-byte hexadecimal message authentication code (MAC) used to verify the message when MACs are being used.

FID H — Authentication Key

Request: Not available.

Response: Optional. Variable length of 16, 32, or 48 bytes. Trailing blanks are removed. This field can be configured in any response and is included if more than the configured number of MAC errors occur. This field can also be included optionally in downloads.

The Authentication Key is a working key generated by the host and provided to the terminal. The Authentication Key is the MAC communications key (KMAC), encrypted under the MAC terminal master key. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

FID I — Data Encryption Key

Request: Not available.

Response: Optional. Variable length of 16, 32, or 48 bytes. This field can be configured in any response and is included if more than the configured number of message encryption errors occur. This field can also be included optionally in downloads.

The Data Encryption Key is a working key generated by the security module and provided to the terminal. The data encryption key is the data encryption communications key (KME) encrypted under the data encryption terminal master key. The Data Encryption Key is also used for full message encryption and configurable message encryption. This key is used to encrypt the Available Balance field. The number and type of keys used determine the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

FID J — Available Balance

Request: Not available.

Response: Optional. Fixed length of 18 bytes. The Available Balance can originate with the authorizer. This field is right-justified and zero filled.

The Available Balance is the amount available to the customer from the account against which a debit or credit card transaction was processed. The Available Balance can also contain the available balance returned from the mobile operator in a mobile top-up transaction. The amount can be encrypted or in the clear. The host can be configured to return balances in this field as follows:

- All transactions
- Purchase, purchase with cash back, preauthorization purchase, and balance inquiry transactions only
- Balance inquiry transactions only

If this FID is configured in other responses, it is zero filled.

FID K — Business Date

Request: Optional. Fixed length of 6 bytes. If this field is not included in the subtotals request, totals for the posting date in the terminal data in the host database are returned.

Response: Optional. Fixed length of 6 bytes. If included, the value is echoed from the request.

The Business Date (YYMMDD) enables terminals to specify an effective posting date for a transaction (other than the current host application transaction log date). This date is recorded in the transaction log at the host. It may vary from the date in the standard header.

FID L — Check Type/Category

Request: Optional. Fixed length of 1 byte. Required for a check verification or check guarantee transaction.

Response: Optional. Fixed length of 1 byte. If included, the value is echoed from the request.

For check verification or check guarantee transactions, the Check Type/Category field indicates the type of check involved. For other financial transactions, this field contains the transaction category.

Financial Transaction Categories

0 = Unspecified check/normal transaction
1 = Sales draft
2 = Representation
3 = Chargeback

Check Types

4 = Personal check for cash
5 = Personal check for purchase
6 = Personal check for purchase and cash
7 = Government
8 = Payroll
9 = Electronic

FID M — PIN Communications Key

Request: Not available.

Response: Optional. Variable length of 16, 32, or 48 bytes. Trailing bytes are removed.

The PIN Communications Key is a working key generated by the host or security module and provided to the terminal. This key is the PIN communications key (KPE), encrypted under the terminal master key.

The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

FID N — Customer ID

Request: Optional. Variable length of 1 to 40 bytes. Trailing blanks are removed.

Response: Optional. Variable length of 1 to 40 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

The Customer ID identifies the customer in a check guarantee or check verification transaction. The value in this field could be a social security number, a drivers license number, or another type of identification.

FID O — Customer ID Type

Request: Optional. Fixed length of 2 bytes. If this FID is not present, the default check ID in the terminal data in the host database is used.

Response: Optional. Fixed length of 2 bytes. If included, the value is echoed from the request.

The Customer ID Type specifies the type of identification used in the Customer ID field. The values in these fields are logged by the host application. Valid values are as follows:

00 = None
01 = Credit card
02 = Drivers license
03 = Checking account number
04 = Debit card
05 = Proprietary check cashing card
06 = State ID number
07 = Social security number
08 = Student ID number
09 = Employee ID
10 = Passport number
12–50 = Reserved for national use
51–75 = Reserved for ISO use
76–99 = Reserved for private use

FID P — Draft Capture Flag

Request: Optional. Fixed length of 1 byte.

Response: Optional. Fixed length of 1 byte. This field in the response represents the draft capture flag used for transactions by the host application.

The Draft Capture Flag can be specified by the terminal. However, the transaction profile kept in the terminal data in the host database overrides the value in this field, unless the transaction profile in the host database indicates that the terminal determines data capture mode for each transaction, in which case the host uses the value in this field. Values for this field would typically originate in data downloaded to the terminal by card prefix ranges. Valid values are as follows:

0 = Authorize only
1 = Authorize and capture

FID Q — Echo Data

Request: Optional. Variable length of 1 to 16 bytes. The data is padded with trailing blanks.

Response: Optional. Variable length of 1 to 16 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

The Echo Data represents data that the terminal requires to be echoed back to it in the response. The host application does not edit this field. It is recorded in the transaction log data and returned in the response to the terminal.

FID R — Card Type

Request: Optional for credit or debit card. Mandatory for mobile top-up transactions using cash. Fixed length of 1 byte. If this field is not submitted, the host application uses a default value for combination cards from its database. If this field is submitted for a non-combination type card, host application ignores it and uses the card type from the card prefix database.

Response: Optional for credit or debit card. Mandatory for mobile top-up transactions using cash. Fixed length of 1 byte. The value in this field is the card type as determined by the host application or the value of N as entered in the request for mobile top-up using cash.

The Card Type enables the terminal to specify the intended usage for a card used as a debit card and a credit card. Valid values are as follows:

C = Credit card
D = Debit card
N = No card type. Used with mobile top-up transactions using cash.

FID S — Invoice Number

Request: Optional. Variable length of 1 to 10 bytes.

Response: Optional. Fixed length of 10 bytes. If included, the value is echoed from the request. This field is left-justified and blank filled.

The Invoice Number enables a terminal to submit a unique stamp to further identify a transaction.

FID T — Invoice Number/Original

Request: Optional. Variable length of 1 to 10 bytes. This field may be required by some interfaces or authorizing entities.

Response: Optional. Fixed length of 10 bytes. If included, the value is echoed from the request. This field is left-justified and blank filled.

The Invoice Number/Original field enables the terminal to uniquely identify a previous transaction that is now being adjusted.

FID U — Language Code

Request: Optional. Fixed length of 1 byte.

Response: Optional. Fixed length of 1 byte. If included in both the request and response, this field echoes the request. If this field was not included in the request, then this field represents the default language code for the terminal from the host database.

The Language Code enables the terminal to override the default language in the terminal data in the host database. It is used in formulating a terminal display response. The terminal can select one of three different language displays. Valid values are as follows:

0 = Table 1
1 = Table 2
2 = Table 3

FID V — Mail/Download Key

Request: Optional. Fixed length of 15 bytes. Required for a read mail request transaction and a mail delivered transaction. Required for downloads except on initial requests.

Response: Optional. Fixed length of 15 bytes. Required for a read mail request transaction and a mail delivered transaction. Required for downloads except on initial requests.

The Mail/Download Key consists of a group of fields that identify mail to be read or marked as delivered, or the type of download to be performed.

Concerning mail, this group identifies a mail category (user-defined), the type of mail access desired (any mail, only undelivered mail, specific mail), a flag used for mail broadcast processing, and a mail ID.

The mail ID serves to identify a specific piece of mail or the ID of the last mail read. The format is as follows:

Category	X(2)
Access Code	9(1)
Processing Flag	X(2)
Mail Date	9(6)
Mail ID	9(4)

Concerning downloading, this group specifies either a full or partial download. If full, this group identifies whether this is the initial request or a continuation request (when the download spans several request/response pairs). If partial, this group also identifies which download field is requested. The format is as follows:

Category	X(2)
Access Code	X
Processing Flag	X(2)
Filler	X(10)

Note: In continuation requests (i.e., read next mail or continue download), the terminal should echo the value contained in the previous response.

FID W — Mail/Download Text

Request: Optional. Required for a send mail transaction request. Variable length of 1 to 449 bytes.

Response: Optional. Variable length of 1 to 449 bytes for mail transactions and 1 to 957 bytes for download text. With responses, only the text of the mail itself is used. If no mail was found, this field is not included in the response, even if the customer configuration supports read mail responses.

The Mail/Download Text field consists of a group of fields representing the text of a send mail request (from the terminal to the host) or the download fields from the host.

Concerning mail, this group contains a maximum of 449 bytes and identifies the destination DPC number (when more than one exists) and the text itself. The format is as follows:

Position	Length	Description
01	**1**	**Destination DPC Number Code** The values for the DPC Number Code are 0, 1, and 2. These values map to different four-character DPC numbers listed in the host application database.
02–449	**448**	**Mail Text (variable-length, 448 bytes maximum)** Text of the mail message.

Concerning downloading, this group contains the fields being downloaded from the host. The group contains a maximum of 956 bytes of download data. It identifies the destination DPC number (when more than one exists) and the text itself.

The format is as follows:

Position	Length	Description
01	**1**	**Destination DPC Number Code** This field is not used for downloads and contains a value of 0.
02–957	**956**	**Download Text (variable-length, 956 bytes maximum)** Text being downloaded.

FID X — ISO Response Code

Request: Not available.

Response: Optional. Fixed length of 3 bytes.

The ISO equivalent of the host response code found in the message header. Used to inform the terminal of transaction processing results.

The ISO equivalent can be a three-digit value based on the International Organization for Standardization (ISO) 8583-1993 standard or a two-digit value based on the 8583-1987 standard. If used, the two-digit value must be left-justified with a trailing blank.

FID Y — Postal (ZIP) Code

Request: Optional. Fixed length of 9 bytes.

Response: Optional. Fixed length of 9 bytes. If included, the value is echoed from the request.

The Postal (ZIP) Code field contains the ZIP code of the cardholder's billing address. The ZIP code should be either five or nine digits in length. If it is five digits in length, the digits should be left-justified and the remaining positions are space filled.

FID Z — Address Verification Status Code

Request: Not available.

Response: Optional. Fixed length of 1 byte.

For responses, this FID contains the address verification status code. The address verification status code identifies the results of comparing address verification information received in the transaction and address verification information contained in the processor's database. Basic values are as follows. Note that if a transaction is sent to an interchange for processing, the response message may contain other interchange-specific values.

A = Address. Addresses matched, but ZIP codes did not match.
E = The transaction was either not eligible for address verification or an error occurred while attempting to process the message.
N = Error. Neither the addresses nor the ZIP codes matched.
R = Retry. The primary and secondary authorizers were either unavailable or they declined the transaction and address verification was not performed by the host application.
S = Service not supported. The host application authorized the transaction, but does not support address verification.
U = Unavailable. Address information was not available to the processor performing address verification.
W = Whole ZIP. The nine-digit ZIP code matched, but the address did not match.
X = Exact. Both the addresses and the nine-digit ZIP codes matched.
Y = Yes. Both the addresses and the five-digit ZIP codes matched.
Z = ZIP. The five-digit ZIP codes matched, but the addresses did not match.
b = Address verification information was not included in the transaction.
0 = Address verification information was included in the transaction, but was not verified. This code is used by transactions to be verified by either a host or an interchange. In addition, transactions to be verified by the host application that are declined before address verification can be performed also carry this code.

FID a — Optional Data

Request: Optional. Variable length of 1 to 250 bytes.

Response: Optional. Variable length of 1 to 250 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

The Optional Data field allows the terminal to exchange any type of optional data with the host application. Typically, this field is used to indicate product codes, quantities, and amounts within a total purchase amount.

FID b — PIN/Customer

Request: Optional. Variable length of 1 to 16 bytes. This field is in PIN/PAD or PIN/PAN PIN block format. The PIN PAD character is 1 byte. Derived unique key per transaction (DUKPT) encrypted PIN blocks are also carried in this field.

Response: Not available.

The PIN/Customer field contains the customer-entered PIN in a clear or encrypted form, depending on whether the terminal supports PIN encryption.

FID c — PIN/Supervisor

Request: Optional. Variable length of 1 to 16 bytes. The PIN PAD character is 1 byte. Typically, this field is submitted only in certain transaction requests, like returns and adjustments. In these cases, host application is also configured to indicate that supervisor security is to be applied to these transactions. This field is in PIN/PAD or PIN/PAN PIN block format. Derived unique key per transaction (DUKPT) encrypted PIN blocks are also carried in this field.

Response: Not available.

The PIN/Supervisor field contains the supervisor-entered PIN in a clear or encrypted form, depending on whether the terminal supports PIN encryption.

FID d — Retailer ID

Request: Optional. Variable length of 1 to 12 bytes. The ID value is padded with trailing blanks.

Response: Optional. Variable length of 1 to 12 bytes. If included, the value is echoed from the request. Trailing blanks are removed.

The Retailer ID field contains the ID assigned to a merchant group by organizations such as MasterCard, VISA, or American Express.

FID e — POS Condition Code

Request: Optional. Fixed length of 2 bytes.

Response: Optional. Fixed length of 2 bytes. If included, the value is echoed from the request.

The POS Condition Code field further describes the transaction being submitted. Valid values are as follows:

00 = Normal presentment
01 = Customer not present
02 = Unattended terminal able to retain card
03 = Merchant suspicious
04 = Electronic cash register interface
05 = Customer present but card not present
06 = Preauthorization request
07 = Telephone device request
08 = Mail or telephone order
09 = Security alert
10 = Customer identity verified
11 = Suspected fraud
12 = Security reasons
13 = Representment of item
14 = Public utility terminal
15 = Customer terminal (Home terminal)
16 = Administration terminal
17 = Returned item (Chargeback)
18 = No check in envelope/all returned
19 = Deposit out-of-balance/all returned
20 = Payment out-of-balance/all returned
21 = Manual reversal

22 = Terminal error/counted
23 = Terminal error/not counted
24 = Deposit out-of-balance/applied contents
25 = Payment out-of-balance/applied contents
26 = Withdrawal had error/reversed
27 = Unattended terminal unable to retain card
28–40 = Reserved for ISO use
41–50 = Reserved for National use
51–99 = Reserved for Private use

FID f — PIN Length or Receipt Data

Request: Optional. Variable length of 1 to 2 bytes representing 0–16 in binary.

Response: Optional. Variable length of 1 to 200 bytes. Trailing blanks are removed.

FID f identifies different data, depending on whether it is associated with a request or a response message. In a request message, FID f identifies the PIN length. In a response message, FID f identifies the receipt data.

The PIN Length field contains the actual length of the PIN entered at the terminal.

This field can optionally contain 1 to 400 bytes of generic marketing message from the Mobile Operator File (MOF) or a marketing message from the mobile operator host.

The Receipt Data field applies only to situations where the customer has commissioned the host application vendor to format receipts. In this case, this field contains the receipt data.

FID g — Response Display

Request: Not available.

Response: Optional. Variable length 1 to 48 bytes. Trailing blanks are removed.

The Response Display field represents a 48-character display that explains the host response code contained in the standard header or the ISO response code contained in FID X. This field is extracted from one of the three host language tables, depending on the default language code stored for the terminal in the host database or the language code included in the request.

FID h — Sequence Number

Request: Optional, except for employee subtotals transaction requests for which the Sequence Number is required if totals for a specific batch are requested. Fixed length of 10 bytes.

Response: Optional. Fixed length of 10 bytes. When this field is included in the request message and the number is found to be valid, the value is echoed in the response if it is to be included. However, this field is returned in a response, even if it is not configured to be included, if the sequence number is included in the request, and the host finds the number to be invalid. If this field is included in the response but not in the request, the host application generates the sequence number internally.

The Sequence Number field consists of a group of fields. The purpose of this group of fields is to help ensure that the host receives and processes every transaction only once. The structure is shown below. Included for each field is its position in the element, its length, and a description of its contents. This group of fields consists of the following:

Position	Length	Description
01–03	**3**	**Shift Number** The Shift Number ranges from 001 to 999. The Shift Number increases by one digit whenever the shift is closed. It rolls to 001 after the 999th shift is closed.

Position	Length	Description
04–06	**3**	**Batch Number** The Batch Number ranges from 001 to 999. The Batch Number increases by one digit whenever the batch is closed. It rolls to 001 after the 999th batch is closed or when the shift is closed. Whenever the shift closes, the batch is also expected to close.
07–09	**3**	**Seq #** The Seq # ranges from 001 to 999. It is a unique number for the current batch. When the Batch Number changes, the Seq # is set to 001. This portion of FID h is called Seq # so as not to confuse it with the name of FID h, which is Sequence Number.
10	**1**	**Reset Flag** The Reset Flag indicates whether the terminal or the host is responsible for determining the correct Shift Number, Batch Number, and Seq #. Only the terminal can set the Reset Flag. Valid values are as follows: 0 = Do not reset the sequence number. 1 = Reset the sequence number. When the terminal submits the Sequence Number, the host checks the Shift Number, Batch Number, and Seq # against the expected values. For information on the actions of the host if it receives a Shift Number, Batch Number, or Seq # it does not expect, see section 4.

FID i — Sequence Number/Original

Request: Optional. Fixed length of 9 bytes. Required for preauthorization purchase transactions and preauthorization purchase completion transactions. Without this field, holds are not removed from customer accounts efficiently.

Response: Optional. Fixed length of 9 bytes. If included, the value is echoed from the request.

The Sequence Number/Original field enables the terminal to optionally identify the sequence number of a previous transaction. For more information on this field, see FID h. FID i has the same format as positions 1 through 9 of FID h.

FID j — State Code

Request: Optional. Fixed length of 2 alphanumeric bytes.

Response: Not available.

The State Code is associated with the transaction request. Valid values are defined by the check verification provider.

Note: For electronic check authorization transactions sent to Visa, the state code entered must be a valid Visa state code.

FID k — Birth Date/Drivers License/Terminal Location

Request: Optional. Fixed length of 6 bytes.

Response: Optional. Variable length of 1 to 25 bytes. Trailing blanks are removed.

In requests, the Birth Date/Drivers License/Terminal Location field contains the birth date (MMDDYY) of the customer associated with the transaction. If no day was indicated on the customer ID containing the birth date, or if the date is an expiration date, the format is MMYY.

In responses, this field contains the terminal location as indicated in the terminal data in the host database. This field is used primarily for Regulation E purposes.

FID l — Totals/Batch

Request: Optional. Fixed length of 75 bytes.

Response: Optional. Fixed length of 75 bytes. Totals can be returned in any response, not just a balancing transaction.

The Totals/Batch field consists of a group of fields representing batch totals as accumulated by the terminal. This field includes a shift and batch ID, along with counts and amounts of debits, credits, and adjustments. The host application saves these totals and the totals accumulated by the host in the transaction log file. These totals contain a sign character (+ or –) in the first byte of each amount field. The format is as follows:

Shift number	9(3)
Batch number	9(3)
Number of debits in batch	9(4)
Amount of debits in batch	S9(16)v99
Number of credits in batch	9(4)
Amount of credits in batch	S9(16)v99
Number of adjustments in batch	9(4)
Amount of adjustments in batch	S9(16)v99

FID m — Totals/Day

Request: Optional. Fixed length of 75 bytes.

Response: Optional. Fixed length of 75 bytes. Totals can be returned in any response, not just a balancing transaction.

The Totals/Day field consists of a group of fields representing terminal day totals as accumulated by the terminal. This field includes a shift and batch count, along with counts and amounts of debits, credits, and adjustments. The host application saves these totals and the totals accumulated by the host in the transaction log file. These totals contain a sign character (+ or –) in the first byte of each amount field. The format is as follows:

Number of shifts in day	9(3)
Number of batches in day	9(3)
Number of debits in day	9(4)
Amount of debits in day	S9(16)v99
Number of credits in day	9(4)
Amount of credits in day	S9(16)v99
Number of adjustments in day	9(4)
Amount of adjustments in day	S9(16)v99

FID n — Totals/Employee

Request: Not available.

Response: Optional. Fixed-length of 121 bytes.

The Totals/Employee field consists of a group of fields representing employee totals as accumulated by the terminal. This field includes a shift and batch ID, along with counts and amounts of debits, credits, adjustments, cash backs, and checks.

These totals contain a sign character (+ or –) in the first byte of each amount field. The format is as follows:

Current shift number	9(3)
Current batch number	9(3)
Number of debits for employee	9(4)
Amount of debits for employee	S9(16)v99
Number of credits for employee	9(4)
Amount of credits for employee	S9(16)v99
Number of adjustments for employee	9(4)
Amount of adjustments for employee	S9(16)v99
Number of cash outs for employee	9(4)
Amount of cash outs for employee	S9(16)v99
Number of checks for employee	9(4)
Amount of checks for employee	S9(16)v99

FID o — Totals/Shift

Request: Optional. Fixed length of 75 bytes.

Response: Optional. Fixed length of 75 bytes. Totals can be returned in any response, not just a balancing transaction.

The Totals/Shift field consists of a group of fields representing terminal shift totals as accumulated by the terminal. This field includes a shift ID and batch count, along with counts and amounts of debits, credits, and adjustments. The host

application saves these totals and the host-accumulated totals in the transaction log file. These totals contain a sign character (+ or –) in the first byte of each amount field. The format is as follows:

Number of shifts in day	9(3)
Number of batches in shift	9(3)
Number of debits in shift	9(4)
Amount of debits in shift	S9(16)v99
Number of credits in shift	9(4)
Amount of credits in shift	S9(16)v99
Number of adjustments in shift	9(4)
Amount of adjustments in shift	S9(16)v99

FID q — Track 2/Customer

Request: Optional. Variable length of 1 to 40 bytes. Either this field or the Track 1/Customer field (FID 2) is required for the following transactions.

Normal purchase
Preauthorization purchase
Preauthorization purchase completion
Mail or telephone order
Merchandise return
Cash advance
Card verification
Balance inquiry
Purchase with cash back
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment

The customer determines when or if Track 2 data is required in the host application database. The host checks that all financial transactions contain either the Track 2/Customer field (FID q) or the Track 1/Customer field (FID 2).

Response: Optional. Variable length of 1 to 40 bytes. If included, the value is echoed from the request.

The Track 2/Customer field consists of a group of fields representing the customer's Track 2. This data can be entered manually, obtained from the terminal's card swipe reader, or obtained from a microprocessor chip on an integrated circuit card (ICC). The format indicates which of these methods was used. The format is as follows:

For manually entered data:	X(29)
Entry ID (M)	X(1)
PAN (variable-length, 19 bytes maximum)	X(19)
Separator character (=)	X(1)
Expiration date (YYMM)	9(4)
Member number	9(3)
End sentinel (?)	X(1)
For swiped data:	X(40)
Start sentinel (;)	X(1)
PAN (variable-length, 19 bytes maximum)	X(19)
Separator character (=)	X(1)
Expiration date	9(4)
Service Code	9(3)
Discretionary data (14 bytes maximum)	X(14)
End sentinel (?)	X(1)
For contactless magnetic stripe transaction data:	X(40)
Start sentinel (;)	X(1)
PAN (variable-length, 19 bytes maximum)	X(19)
Separator character (=)	X(1)
Expiration date	9(4)
Service Code	9(3)
Discretionary data (14 bytes maximum) Note: This field will contain the information necessary for dynamic card verification (dynamic card verification value, application transaction counter, and unpredictable number, depending on the contactless program supported).	X(14)
End sentinel (?)	X(1)
For chip read data (if EMV Tag 57 is present):	X(39)
Start sentinel (;)	X(1)
Track 2 equivalent data (**EMV Tag:** 57)	X(37)
End sentinel (?)	X(1)
For chip read data (if EMV Tag 57 is not present):	X(26)
Entry ID (M)	X(1)
PAN (variable-length, 19 bytes maximum) (**EMV Tag:** 5A)	X(19)
Separator character (=)	X(1)

Expiration date (YYMM) (**EMV Tag:** 5F24)	9(4)
Service Code	9(3)
End sentinel (?)	X(1)

Note: For EMV transactions, the Application PAN Sequence Number (EMV Tag: 5F34) is carried in the EMV Additional Request Data field (FID 6, subFID P). EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this FID description for reference purposes only.

FID r — Track 2/Supervisor

Request: Optional. Variable length of 1 to 40 bytes.

Response: Optional. Variable length of 1 to 40 bytes. If included, the value is echoed from the request.

The Track 2/Supervisor field consists of a group of fields representing the supervisor Track 2. Supervisor Track 2 data is typically submitted only in certain transaction requests such as returns or adjustments. In these cases, the host is also configured to indicate that the supervisor security is to be applied to these transactions. The format of the Track 2/Supervisor data is the same as that of the Track 2/Customer data carried in FID q.

FID s — Transaction Description

Request: Not available.

Response: Optional. Variable length of 1 to 24 bytes.

The Transaction Description field is a 24-character description of the transaction for receipt purposes.

FID t — PIN Pad Identifier

Request: Optional. Fixed length of 16 bytes. If not supplied, the acceptor uses the Terminal ID in the message header.

Response: Optional. Fixed length of 16 bytes. The value is echoed from the PIN pad identifier received in the request.

The PIN pad identifier is the logical identifier of the PIN pad at the acquiring terminal and is unique in the accepting host environment.

FID u — Acceptor Posting Date

Request: Not available.

Response: Optional. Fixed length of 6 bytes. If this date is not supplied, the acquirer Merchant Link process uses its own posting date, resulting in settlement totals that may not agree.

The acceptor posting date (YYMMDD) is the date of the transaction log file to which the acceptor logged the transaction. The acceptor returns this field so that transactions can be matched between the acquirer and acceptor, and reconciliation totals can be accumulated regardless of individual cutover configurations. This field can also be returned by the acceptor on transactions originated at POS devices.

FID 0 — AMEX Data Collection

Request: Optional. Variable-length of 46 to 118 bytes, depending on the standard industry format being used. The data length for each specific standard industry format is fixed.

Response: Optional. Variable length of 46 to 118 bytes, depending on the standard industry format being used. The data length for each specific standard industry format is fixed. This field is not returned in a response if the transaction is declined.

The AMEX Data Collection field is used to capture transactions originating from American Express cardholders. American Express has defined categories under which transactions are placed. These categories are as follows:

- Auto rental
- Lodging
- Restaurant
- General retail
- Oil

For each of these categories, American Express requires different data to be sent from the device. Therefore, American Express has set forth standard industry formats for each category to ensure the data they require is captured by the device and sent to the host. The host is able to recognize these standard industry formats and, subsequently, capture and process transactions sent using them. In conjunction with this field, the customer must set up Standard Industrial Classification (SIC) Codes or Merchant Category Codes for the terminal in the host database. SIC and Merchant Category Codes identify the merchant's line of business. For the American Express standard industry formats, refer to appendix B.

FID 1 — PS2000 Data

Request: Optional. Fixed length of 24 bytes.

Response: Optional. Fixed length of 24 bytes.

The PS2000 Data field consists of a group of fields used with the Visa PS2000 program. This field contains PS2000 data for both request and response messages. The terminal is responsible for correctly filling in any of the fields that are required in a given request, placing default values in the other fields, and parsing any of the fields out of the response message that need to be printed on a receipt or displayed. The format for this field is as follows:

Authorization characteristics indicator	PIC X(1)
Transaction identifier	PIC X(15)
Validation code	PIC X(4)
Market-specific data fields	
Market-specific data identifier	PIC X(1)
Duration	PIC 9(2)
Prestigious property indicator	PIC X(1)

FID 2 — Track 1/Customer

Request: Optional. Variable length of 1 to 82 bytes. Either this field or the Track 2/Customer field (FID q) is required for the following transactions:

Normal purchase
Preauthorization purchase
Preauthorization purchase completion
Mail or telephone order
Merchandise return
Cash advance
Card verification
Balance inquiry
Purchase with cash back
Purchase adjustment
Merchandise return adjustment
Cash advance adjustment
Cash back adjustment

The customer determines when or if Track 1 data is required in the host application database. The host checks that all financial transactions contain either the Track 2/Customer field (FID q) or the Track 1/Customer field (FID 2).

Response: Optional. Variable length of 1 to 82 bytes. If included, the value is echoed from the request.

The Track 1/Customer field consists of a group of fields representing the customer's Track 1. This data is obtained from the terminal's card swipe reader. The format for customer Track 1 data, organized in ISO Standard Format, is as follows:

Field	Description
Start sentinel	1 character (%)
Format code	1 character (B for credit cards)
Identification (PAN)	Up to 19 digits (variable length)
Field separator	1 character (^)
Country code	3 digits (only present when the PAN starts with 59)
Name	Up to 26 characters (variable length)
Field separator	1 character (^)
Expiration date	4 digits (YYMM)
Service code	3 digits

Field	Description
Discretionary data	Up to 21 characters (variable length)
End sentinel	1 character (?)
Longitudinal redundancy check character	1 character

Note: For contactless transactions, the Discretionary Data field will contain the information necessary for dynamic card verification (dynamic card verification value, application transaction counter, and unpredictable number, depending on the contactless program supported).

FID 3 — Track 1/Supervisor

Request: Optional. Variable length of 1 to 82 bytes. Supervisor Track 1 is typically submitted only in certain transaction requests such as returns and adjustments. In these cases, the host is also configured to indicate that supervisor security is to be applied to these transactions.

Response: Optional. Variable length of 1 to 82 bytes. If included, the value is echoed from the request.

The Track 1/Supervisor field consists of a group of fields representing the supervisor Track 1. This data is obtained from the terminal's card swipe reader. The supervisor Track 1 is typically submitted only in certain transaction requests, such as returns or adjustments. In these cases, the host is also configured to indicate that the supervisor security is to be applied to these transactions. The format for supervisor Track 1 data, organized in ISO Standard Format, is as follows:

Field	Description
Start sentinel	1 character (%)
Format code	1 character (B for credit cards)
Identification (PAN)	Up to 19 digits (variable length)
Field separator	1 character (^)
Country code	3 digits (only present when the PAN starts with 59)
Name	Up to 26 characters (variable length)
Field separator	1 character (^)
Expiration date	4 digits (YYMM)
Service code	3 digits

Field	Description
Discretionary data	Up to 21 characters (variable length)
End sentinel	1 character (?)
Longitudinal redundancy check character	1 character

FID 4 — Industry Data

Request: Optional. Variable length of 156 to171 bytes.

Response: Optional. Variable length of 156 to171 bytes. If included, the value is echoed from the request.

The Industry Data field contains information associated with lodging and vehicle rental. The format is as follows:

Industry Type	PIC X(2)
Industry Data	PIC X(169), variable length

These fields are further described below.

Position	Length	Description
01–02	**2**	**Industry Type** A code that identifies the type of industry data. Valid values are as follows: LG = Lodging VR = Vehicle rental
03–171	**169**	**Industry Data (variable-length, 154–169 bytes)**
03–156	**154**	**Lodging** The following fields are used for lodging data.
03–08	**6**	**Arrival Date (YYMMDD)** The date the customer checked in. For a no-show or advanced lodging transaction, this is the scheduled arrival date.
09–14	**6**	**Departure Date (YYMMDD)** The date the customer checked out.

Position	Length	Description
15–18	**4**	**Total Room Nights** The total number of room nights during the lodging stay.
19–30	**12**	**Room Rate** The daily room charges exclusive of taxes and fees. Two decimal places are implied.
31–42	**12**	**Room Tax** The daily room tax. Two decimal places are implied.
43–54	**12**	**Phone Charges** The total amount of charges for all phone calls. Two decimal places are implied.
55–66	**12**	**Laundry Charges** The total amount of laundry and dry cleaning charges. Two decimal places are implied.
67–78	**12**	**Gift Shop Charges** The total amount of gift shop and specialty shop charges. Two decimal places are implied.
79–90	**12**	**Bar Charges** The total amount of bar and in-room mini-bar charges. Two decimal places are implied.
91–102	**12**	**Other Charges** The total amount of other charges associated with the lodging stay. Two decimal places are implied.
103–114	**12**	**Total Tax Amount** The total amount of sales tax or value-added tax on the total purchase. Two decimal places are implied.
115–129	**15**	**Property Phone Number** Identifies the specific lodging property location by its local phone number.

Position	Length	Description
130–144	**15**	**Customer Service Phone Number** The phone number used to resolve cardholder questions and disputes.
145–154	**10**	**Folio Number** The merchant's internal invoice or billing ID reference number.
155	**1**	**Fire Safety Act Indicator** A code that identifies whether the facility is in compliance with the Hotel and Motel Fire Safety Act of 1990 (PL101-391), or similar legislation. Valid values are as follows: Y = Yes, the facility is in compliance. N = No, the facility is not in compliance.
156	**1**	**No Show Indicator** A code indicating whether the individual showed up after making a reservation for lodging. Valid values are as follows: 0 = Not applicable. 1 = No show. Transaction amount is due.
03–171	**169**	**Vehicle Rental** The following fields are used for vehicle rental data.
03–31	**29**	**Renter Name** The name of the individual making the vehicle rental agreement.
32–35	**4**	**Rental Class ID** The classification of the vehicle rented, such as midsize or luxury.
36–41	**6**	**Rental Date (YYMMDD)** The date the customer picked up the vehicle from the rental agency.

Position	Length	Description
42–59	**18**	**Rental City** The city where the customer picked up the vehicle.
60–62	**3**	**Rental State** The state or province where the customer picked up the vehicle. This field must contain a valid U.S. state code if the rental country is USA.
63–65	**3**	**Rental Country** The country where the customer picked up the vehicle. This field must contain a valid alphabetic ISO country code.
66–71	**6**	**Return Date (YYMMDD)** The date the customer returned the vehicle.
72–89	**18**	**Return City** The city where the customer returned the vehicle.
90–92	**3**	**Return State** The state or province where the customer returned the vehicle. This field must contain a valid U.S. state code if the rental country is USA.
93–95	**3**	**Return Country** The country where the customer returned the vehicle. This field must contain a valid alphabetic ISO country code.
96–105	**10**	**Return Location ID** The code, address, phone number, or other identifier used to identify the location where the customer returned the vehicle.
106–109	**4**	**Days Rented** The number of days the vehicle was rented.

Position	Length	Description
110–121	**12**	**Daily Rental Rate** The daily rental rate, exclusive of taxes and fees. Two decimal places are implied.
122–133	**12**	**Extra Charges** The total amount of extra charges associated with the vehicle rental. Two decimal places are implied.
134–145	**12**	**Total Tax Amount** The total amount of sales tax or value-added tax on the total purchase. Two decimal places are implied.
146–160	**15**	**Customer Service Phone Number** The phone number used to resolve cardholder questions and disputes.
161–169	**9**	**Agreement Number** The invoice number of the original rental agreement.
170	**1**	**Tax Exempt Indicator** A code indicating whether the goods or services were tax exempt. Valid values are as follows: 0 = Not applicable 1 = Tax exempt
171	**1**	**No Show Indicator** A code indicating whether the individual showed up after reserving a vehicle for rental. Valid values are as follows: 0 = Not applicable. 1 = No show. Transaction amount is due.

FID 6 — Product SubFIDs

Request: Optional. Variable length.

Response: Optional. Variable length.

This Product SubFIDs field consists of subordinate optional data subfields. For information about these subfields, see the topic "Optional Data Subfields — FID 6" later in this section.

FID 7 — Product SubFIDs

Request: Optional. Variable length.

Response: Optional. Variable length.

This Product SubFIDs field consists of subordinate optional data subfields. For information about these subfields, see the topic "Optional Data Subfields — FID 7" later in this section.

FID 8 — Product SubFIDs

Request: Optional. Variable length.

Response: Optional. Variable length.

This Product SubFIDs field consists of subordinate optional data subfields. For information about these subfields, see the topic "Optional Data Subfields — FID 8" later in this section.

FID 9 — Customer SubFIDs

Request: Optional. Variable length.

Response: Optional. Variable length.

The Customer SubFIDs field is reserved for future use by customers.

Optional Data Subfields — FID 6

The optional data subfields for FID 6 (Product SubFIDs) are summarized in a table below, followed by individual descriptions of the subfields.

Note: The information in FIDs 6 through 9 are contained in subfields. These subfields are included in the request or response message when FIDs 6 through 9 are specified in the host configuration. Specifying FIDs 6 through 9 in the host configuration specifies all subfields for FIDs 6 through 9, respectively. FIDs 6, 7, and 8 are reserved for product use. FID 9 is reserved for customer use.

Summary Table

The subfields for FID 6 are described below according to their subfield identifiers (SFIDs). The table lists the SFID, the picture clause and length of the subfield in the message, and its associated field name. In addition, a check mark (✓) appears in the RQST and RESP columns if a subfield is available for requests and responses, respectively.

SFID	Picture	Length	Field Name	RQST	RESP
A	PIC X(12)	12 bytes	Host original data	✓	✓
B	PIC X(4)	4 bytes	Manual card verification data (CVD)—customer	✓	✓
C	PIC X(4)	4 bytes	Manual card verification data (CVD)—administrative	✓	✓
D	PIC X(876)	30–876 bytes	Purchasing card or fleet card data	✓	
E	PIC 999	3 bytes	POS entry mode	✓	✓
F	PIC X(2)	1–2 bytes	Electronic commerce data	✓	
G	PIC X	1 byte	Visa commercial card type indicator		✓
H	PIC X(2)	2 bytes	CVD presence indicator and CVD result	✓	✓
I	PIC 9(3)	3 bytes	Transaction currency code	✓	✓

SFID	Picture	Length	Field Name	RQST	RESP
J	PIC X(32)	32 bytes	Cardholder certificate serial number	✓	
K	PIC X(32)	32 bytes	Merchant certificate serial number	✓	
L	PIC X(80)	80 bytes	XID/trans stain	✓	
N	PIC 9(4)	4 bytes	Message reason code	✓	
O	PIC X(136)	136 bytes (variable length)	EMV request data	✓	
P	PIC X(16)	64 bytes (variable length)	EMV additional request data	✓	
Q	PIC X(36)	64 bytes (variable length)	EMV response data		✓
R	PIC X(82) for requests PIC X(258) for responses	258 bytes (variable length)	EMV Reversal Data/EMV Additional Response Data	✓	✓
S	PIC X(63)	63 bytes	Stored value data	✓	✓
T	PIC X(23)	23 bytes	Key serial number and descriptor	✓	
U	PIC X(16)	16 bytes	Transaction subtype data	✓	
V	PIC X	1 byte	Authentication collection indicator	✓	✓
W	PIC X	1 byte	CAVV/AAV result code		✓
X	PIC X(6)	6 bytes	Point of service data	✓	
Y	PIC X(202)	2–202 bytes	Authentication data	✓	✓
Z	PIC X	1 byte	Card verification flag 2	✓	✓
b	PIC X(39)	39 bytes	Electronic check conversion data	✓	

SFID	Picture	Length	Field Name	RQST	RESP
c	PIC X(64)	64 bytes	MICR data	✓	
d	PIC X(115)	115 bytes	Electronic check callback information	✓	✓
e	PIC X(21)	21 bytes	Interchange compliance data		✓
f	PIC X	1 byte	Response source or reason code		✓
g	PIC X(4)	4 bytes	POS merchant data	✓	
h	PIC X(6)	6 bytes	System Trace Audit Number (STAN)		✓
i	PIC X(12)	12 bytes	Retrieval Reference Number		✓
j	PIC X(4)	4 bytes	Debit Network/Sharing Group ID		✓
k	PIC X(2)	2 bytes	Card Level Results		✓
l	PIC X(120)	20–120 bytes	Healthcare/Transit Data	✓	✓
m	PIC X(95)	19–95 bytes	Healthcare Service Data	✓	✓
n	PIC X	1 byte	Error Flag		✓
o	PIC X(300)	3-300 bytes	American Express Additional Data	✓	
q	PIC X(162)	8–162 bytes	EMV Supplementary Request Data	✓	
r	PIC X(150)	7–150 bytes	Auto-Substantiation Data	✓	

SFID A — Host Original Data

Request: Optional. Fixed length of 12 bytes.

Response: Optional. Fixed length of 12 bytes. If included, the value is echoed from the request.

The Host Original Data field represents the time (hhmmsshh) and date (MMDD) of the original transaction.

SFID B — Manual CVD—Customer

Request: Optional. Fixed length of 4 bytes.

Response: Optional. Fixed length of 4 bytes. If included, the value is echoed from the request.

The Manual CVD—Customer field contains the manually entered card verification digits from the customer's card.

If this field contains a 3-digit American Express card security code (CSC), it must be left-justified and followed by a blank.

SFID C — Manual CVD—Administrative

Request: Optional. Fixed length of 4 bytes.

Response: Optional. Fixed length of 4 bytes. If included, the value is echoed from the request.

The Manual CVD—Administrative field contains the manually entered card verification digits from the administrative card.

If this field contains a 3-digit American Express card security code (CSC), it must be left-justified and followed by a blank.

SFID D — Purchasing Card or Fleet Card Data

Request: Optional. Variable length of 30–876 bytes, depending on the data type.

Response: Not applicable.

The Purchasing Card or Fleet Card Data field contains purchasing or fleet card information. The format of this subfield is as follows:

Card Type	PIC X
Customer Identification Number	PIC X(17)
Tax Amount	PIC 9(12)
Purchase Card or Fleet Card Data	PIC X(846)

Each of these fields is further defined below.

Position	Length	Description
01	**1**	**Card Type** A code indicating the type of card data present. Valid values are as follows: A = American Express purchasing card C = MasterCard fleet card F = Visa fleet card M= MasterCard purchasing card level 3 V = Visa purchasing card level 3 ␣ = Level 2 purchasing card (where ␣ is a blank)
02–18	**17**	**Customer Identification Number** The customer identification number.
19–30	**12**	**Tax Amount** The sales tax assessed and included in the transaction.
31–876	**846**	**MasterCard Purchasing Card Level 3 Data** The following fields contain MasterCard purchasing card level 3 data that redefines the Purchase Card or Fleet Card Data field when the Card Type field is set to a value of M.

Position	Length	Description
31–39	**9**	**Ship From Code** The postal code from which the items were shipped.
40–48	**9**	**Ship To Code** The postal code to which the items will be delivered.
49–51	**3**	**Destination Country Code** A code indicating the country to which items will be delivered.
52–68	**17**	**Merchant Reference Number** The reference number supplied by the merchant for records management.
69–77	**9**	**Freight Amount** The freight charges portion of the transaction amount.
78–86	**9**	**Duty Amount** The importing fee (duty) assessed for the transaction.
87–98	**12**	**Product Code** The product code for the item.
99–133	**35**	**Description** The description of the purchased item.
134–137	**4**	**Quantity** The number of items purchased.
138–140	**3**	**Unit of Measure** A code indicating the unit of measure.
141–149	**9**	**Extended Amount** The total amount of items purchased.

Position	Length	Description
150	**1**	**Debit or Credit Indicator** A code indicating whether the extended amount is a debit or a credit. Valid values are as follows: C = Credit D = Debit
151	**1**	**Discount Indicator** A flag indicating whether a discount was applied to the purchase amount. Valid values are as follows: Y = Yes, a discount was applied to the purchase amount. N = No, a discount was not applied to the purchase amount.
152–160	**9**	**Discount Amount** The amount of the discount applied to the purchase amount.
161	**1**	**Net Gross Indicator** A flag indicating whether the amount includes a tax. Valid values are as follows: Y = Yes, a tax is included in the amount. N = No, a tax is not included in the amount.
162	**1**	**Sales Tax Indicator** A flag indicating whether the amount includes sales tax. Valid values are as follows: Y = Yes, sales tax is included in the amount. N = No, sales tax is not included in the amount.
163–167	**5**	**Value-Added Tax Rate** The value-added tax rate. Five decimal places are implied.
168–171	**4**	**Value-Added Tax Type** The type of value-added tax.

Position	Length	Description
172–180	**9**	**Value-Added Tax Amount** The amount of the value-added tax.
181	**1**	**Alternate Tax Indicator** A flag indicating whether an alternate tax is included in the purchase amount. Valid values are as follows: Y = Yes, an alternate tax is included in the purchase amount. N = No, an alternate tax is not included in the purchase amount.
182–190	**9**	**Alternate Tax Amount** The amount of the alternate tax.
191–205	**15**	**Alternate Tax Identifier** The alternate tax ID number.
206	**1**	**User Field 1** Reserved for future use.
207–876	**670**	**User Field 2** Reserved for future use.
31–876	**846**	**Visa Purchasing Card Level 3 Data** The following fields contain Visa purchasing card level 3 data that redefines the Purchase Card or Fleet Card Data field when the Card Type field is set to a value of V.
31–42	**12**	**Product Code** The product code of the item sold.
43–68	**26**	**Description** The description of the item sold.
69–80	**12**	**Commodity Code** The item commodity code.

Position	Length	Description
81–92	**12**	**Quantity** The number of items purchased. Four decimal places are implied.
93–104	**12**	**Unit of Measure** A code indicating the unit of measure.
105–116	**12**	**Unit Cost** The cost for each item. Four decimal places are implied.
117–128	**12**	**Value-Added Tax Amount** The value-added tax amount. Two decimal places are implied.
129–132	**4**	**Value-Added Tax Rate** The value-added tax rate. Two decimal places are implied.
133–144	**12**	**Discount Amount** The amount of the discount applied to the purchase. Two decimal places are implied.
145–156	**12**	**Total Amount** The total amount of items purchased.
157	**1**	**Detail Indicator** A line item detail indicator.
158	**1**	**User Field 3** Reserved for future use.
159–876	**718**	**User Field 4** Reserved for future use.
31–876	**846**	**MasterCard Fleet Card Data** The following fields contain MasterCard fleet card data that redefines the Purchase Card or Fleet Card Data field when the Card Type field is set to a value of C.

Position	Length	Description
31–34	**4**	**Brand Code** The oil company brand code.
35	**1**	**Service Type** The type of service. Valid values are as follows: 1 = Self service 2 = Full service
36–37	**2**	**Fuel Product Code** A two-digit code defined by MasterCard that identifies the product. Valid values are 01 through 29.
38–42	**5**	**Unit Cost** The fuel price per unit. Three decimal places are implied.
43	**1**	**Unit of Measure** The unit of measure. Valid values are as follows: 1 = Gallon 2 = Liter
44–49	**6**	**Quantity** The quantity of fuel purchased. Two decimal places are implied.
50–58	**9**	**Gross Fuel Price** The gross fuel price. Two decimal places are implied.
59–65	**7**	**Odometer** The odometer reading at the time of purchase.
66–82	**17**	**Vehicle Number** The vehicle identification number.
83–99	**17**	**Identification Number** The drivers license number.

Position	Length	Description
100	**1**	**Product Type** A code read from the card track indicating which prompts occur at the POS device. Valid values are as follows: 1 = Prompt for ID number (drivers license number) and odometer reading. 2 = Prompt for vehicle identification number and odometer reading. 3 = Prompt for user-defined driver number and odometer reading. 4 = Prompt for odometer reading only. 5 = No prompt.
101–105	**5**	**Tax Exemption Amount** The tax amount for tax exempt fleets.
106–120	**15**	**Alternate Tax Identifier** A code identifying an alternate tax.
121–122	**2**	**Decline Reason Code** A code indicating the reason the transaction was denied. Valid values are as follows: 01 = Invalid ID number 02 = Invalid drivers license number 03 = Invalid vehicle identification number
123	**1**	**Nonfuel Items** The number of nonfuel items included in the purchase.
124	**1**	**User Field 5** Reserved for future use.
125–876	**752**	**Nonfuel Data (occurs 8 times)** The following fields contain information about nonfuel purchases.

Position	Length	Description
	12	**Product Code** A two-digit code defined by MasterCard that identifies nonfuel products. Valid values are 30 through 99.
	35	**Description** The description of the nonfuel product.
	5	**Quantity** The quantity of the nonfuel item purchased.
	3	**Unit of Measure** The unit of measure of the nonfuel item.
	9	**Extended Amount** The total purchase amount of the nonfuel item.
	1	**Discount Indicator** A flag indicating whether a discount was applied to the purchase amount. Valid values are as follows: Y = Yes, a discount was applied to the purchase amount. N = No, a discount was not applied to the purchase amount.
	9	**Discount Amount** The amount of the discount applied to the price.
	1	**Net Gross Indicator** A flag indicating whether the amount includes a tax. Valid values are as follows: Y = Yes, tax is included in the amount. N = No, tax is not included in the amount.
	5	**Value-Added Tax Rate** The value-added tax rate. Five decimal places are implied.

Position	Length	Description
	4	**Value-Added Tax Type** The type of value added tax applied to the transaction.
	9	**Tax Amount** The tax paid on the nonfuel items.
	1	**Debit or Credit Indicator** A code indicating whether the transaction is a debit or a credit. Valid values are as follows: C = Credit D = Debit
31–876	**846**	**Visa Fleet Card Data** The following fields contain Visa fleet card data that redefines the Purchase Card or Fleet Card Data field when the Card Type field is set to a value of F.
31	**1**	**Purchase Type** A code indicating the type of merchandise purchased. Valid values are as follows: 1 = Fuel purchase 2 = Nonfuel purchase 3 = Fuel and nonfuel purchase
32–33	**2**	**Fuel Type** A two-character code defined by Visa indicating the type of fuel purchased. Valid values are 00 through FF.
34	**1**	**Unit of Measure** A code indicating the unit of measure. Valid values are as follows: G = Gallon I = Imperial gallon K = Kilo L = Liter P = Pound

Position	Length	Description
35–46	**12**	**Quantity** The quantity purchased. Four decimal places are implied.
47–58	**12**	**Unit Cost** The unit cost. Four decimal places are implied.
59–70	**12**	**Gross Fuel Price** The gross fuel price. Four decimal places are implied.
71–82	**12**	**Net Fuel Price** The net fuel price. Four decimal places are implied.
83–94	**12**	**Net Nonfuel Price** The net nonfuel price. Two decimal places are implied.
95–111	**17**	**Vehicle Number** The vehicle number.
112–128	**17**	**Driver Identification Number** The driver identification number.
129–135	**7**	**Odometer** The odometer reading at the time of purchase.
136–139	**4**	**Value-Added Tax Rate** The value-added tax rate. Two decimal places are implied.
140–151	**12**	**Miscellaneous Fuel Tax** The miscellaneous fuel tax amount. Four decimal places are implied.
152–163	**12**	**Other Tax** The other tax amount.

Position	Length	Description
164–175	**12**	**Miscellaneous Nonfuel Tax** The miscellaneous nonfuel tax amount. Two decimal places are implied.
176	**1**	**Service Type** The type of service. Valid values are as follows: 1 = Self service 2 = Full service
177–192	**2**	**Product Code (occurs 8 times)** A two-digit code defined by Visa that identifies a nonfuel product. Valid values are 30 through 99.
193–204	**12**	**Gross Nonfuel Price** The gross nonfuel price.
205	**1**	**Miscellaneous Fuel Tax Exemption Status** The miscellaneous fuel tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
206	**1**	**Miscellaneous Nonfuel Tax Exemption Status** The miscellaneous nonfuel tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
207	**1**	**Federal Nonfuel Excise Tax Exemption Status** The federal excise tax exemption status for nonfuel products. Valid values are as follows: 0 = Nonexempt 1 = Exempt
208–219	**12**	**Federal Nonfuel Excise Tax Amount** The federal excise tax amount for nonfuel products. Two decimal places are implied.

Position	Length	Description
220	**1**	**Federal Fuel Excise Tax Exemption Status** The federal excise tax exemption status for fuel. Valid values are as follows: 0 = Nonexempt 1 = Exempt
221–232	**12**	**Federal Fuel Excise Tax** The federal excise tax amount for fuel products. Two decimal places are implied.
233	**1**	**State Motor Fuel Tax Exemption Status** The state motor fuel tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
234–245	**12**	**State Motor Fuel Tax Amount** The state motor fuel tax amount. Two decimal places are implied.
246	**1**	**County Fuel Sales Tax Exemption Status** The county fuel sales tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
247–258	**12**	**County Fuel Sales Tax Amount** The county fuel sales tax amount. Two decimal places are implied.
259	**1**	**Nonfuel State and Local Sales Tax Exemption Status** The nonfuel state and local sales tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt

Position	Length	Description
260–271	**12**	**Nonfuel State and Local Sales Tax Amount** The nonfuel state and local sales tax amount. Two decimal places are implied.
272	**1**	**County Motor Fuel Tax Exemption Status** The county motor fuel tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
273–284	**12**	**County Motor Fuel Tax Amount** The county motor fuel tax amount. Two decimal places are implied.
285	**1**	**City Fuel Sales Tax Exemption Status** The city fuel tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
286–297	**12**	**City Fuel Sales Tax Amount** The city fuel tax amount. Two decimal places are implied.
298	**1**	**City Motor Fuel Sales Tax Exemption Status** The city motor fuel tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
299–310	**12**	**City Motor Fuel Sales Tax Amount or Local Tax Amount** The city motor fuel tax amount. Two decimal places are implied.

Position	Length	Description
311	**1**	**Secondary State Fuel Tax Exemption Status** The secondary state fuel tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
312–323	**12**	**Secondary State Fuel Tax Amount** The secondary state fuel tax amount. Two decimal places are implied.
324	**1**	**Federal Sales Tax Exemption Status** The federal sales tax exemption status. Valid values are as follows: 0 = Nonexempt 1 = Exempt
325–336	**12**	**Federal Sales Tax Amount or National Tax Amount** The federal sales tax amount or national tax amount. Two decimal places are implied.
337	**1**	**National Tax Included** The national tax included indicator. Valid values are as follows: 0 = Not subject to tax 1 = Subject to tax
338	**1**	**Local Tax Included** The local tax included indicator. Valid values are as follows: 0 = Tax not included 1 = State or provincial tax included 2 = Transaction is not subject to tax

Position	Length	Description
339–358	**20**	**Merchant VAT Registration / Single Business Reference Number** The merchant value-added tax registration number or single business reference number.
359–371	**13**	**Customer VAT Registration Number** The customer value-added tax registration number.
372–388	**17**	**Customer Code / Customer Reference Identifier** The customer code or customer reference identifier.
389–403	**15**	**Message Identifier** The message identifier that is used to link the separate line item detail messages.
404	**1**	**Additional Data Indicator** The additional data indicator. Valid values are as follows: Y = Yes, additional data is provided separately in Visa BASEII Draft Data TC 50 text message transaction. N = No, additional data is not provided.
405–408	**4**	**Summary Commodity Code** The summary commodity code.
409–876	**468**	**User Field 6** Reserved for future use.

SFID E — POS Entry Mode

Request: Optional. Fixed length of 3 bytes.

Response: Optional. Fixed length of 3 bytes.

A code indicating the manner in which transaction data was entered at the POS device. Valid values for the first and second digits are as follows:

00 = Unspecified
01 = Manually
02 = Magnetic stripe
03 = Bar code
04 = OCR
05 = Integrated circuit card
06 = Reserved for ISO use.
07 = Contactless chip card transaction
08–60 = Reserved for ISO use
61–80 = Reserved for national use
81–90 = Reserved for private use
91 = Contactless magnetic stripe transaction (including a Visa MSD transaction containing a cryptogram)
92–99 = Reserved for private use

Valid values for the third digit are as follows:

0 = Unspecified
1 = PIN entry capability
2 = No PIN entry capability
3–5 = Reserved for ISO use
6–7 = Reserved for national use
8–9 = Reserved for private use

For EMV transactions, the first two digits of this field must be set to a value of 05 and the third digit must be set to a value of 0, 1, or 2.

SFID F — Electronic Commerce Flag

Request: Optional. Variable length of 1 to 2 bytes.

Response: Not applicable.

This subfield contains codes to identify electronic commerce, mail or telephone order transactions, and recurring payments. The format is as follows:

Electronic Commerce Flag	PIC X
Recurring Payment Indicator	PIC X

Valid values for the Electronic Commerce Flag are as follows:

0 = Not an electronic commerce transaction
1 = Single mail or telephone order transaction
2 = Recurring mail or telephone order transaction
3 = Mail or telephone order installment payment
4 = Mail or telephone order unknown classification
5 = Secure electronic transaction with cardholder authentication
6 = Encrypted electronic commerce transaction where the merchant is capable of authenticating the cardholder, but was unable to complete the authentication, (e.g., because the issuer or cardholder does not participate in the appropriate authentication program)
7 = Encrypted electronic commerce transaction
8 = Nonsecure electronic commerce transaction
9 = Non-authenticated security transaction; does not comply with secure electronic transaction and the merchant supports secure electronic transactions
S = Internet electronic delivery; valid for AMEX transactions only.
T = Internet physical delivery; valid for AMEX transactions only.

Valid values for the Recurring Payment Indicator, if it is present, are as follows:

0 = Not a recurring payment
1 = Recurring payment

SFID G — Commercial Card Type

Request: Not applicable.

Response: Optional. Fixed length of 1 byte.

The commercial card type. Valid values are as follows:

0 = Noncommercial card or unknown or unspecified card
B = Business card
R = Corporate card
S = Purchasing card

SFID H — Card Verification Digits Presence Indicator and Result

Request: Optional. Fixed length of 2 bytes.

Response: Optional. Fixed length of 2 bytes. If included, the first byte is echoed from the request.

A field indicating whether the CVD/CVD2/CSC is present, and if so, the result of the CVD check. The first byte is the CVD presence indicator. Valid values for this byte are as follows:

0 = CVV2 value is deliberately bypassed or is not provided by the merchant
1 = CVV2 value is present
2 = CVV2 value is on the card, but is illegible
9 = Cardholder states that the card has no CVV2 imprint

The second byte is the CVD result. Valid values for this byte are as follows:

0 = Card verification was not performed because the transaction was denied before card verification processing started.
C = Card verification was performed and the card verification digits (CVD) were invalid. The situation was noted, and transaction processing continued.
D = Card verification was performed and the CVD was invalid. The transaction was denied and the ERR-FLG field was set to C.
J = CVV checking was not performed. The track length was in error. The host database indicates that the transaction should be denied.

K = Card verification was not performed. The track length was in error. The situation was noted and the transaction was referred.

L = CVV checking was not performed. The track length was in error. The host database indicates that processing should continue.

N or ♭ = Authorizing entity has not attempted card verification or could not verify the CVD due to a security device error. (♭ indicates a blank character.)

O = Card verification was not performed. A CVD value was not on the card. Not all cards have a CVD value encoded. The card expiration date must be equal to or greater than an expiration date defined on the Card Prefix File (CPF) to ensure that the CVD field has been encoded. If the card expiration date is equal to or greater than the CPF date, the CVD checks are performed.

P = Card verification was not performed. Either the merchant ignored the CVD on purpose or the user falsely indicated no CVD was on the card.

R = Card verification was performed and the CVD was invalid. The situation was noted and the transaction should be referred.

U = Issuer has not certified or has not provided the encryption keys to the switch.

Y = Card verification was performed and the CVD was valid.

SFID I — Transaction Currency Code

Request: Optional. Fixed length of 3 bytes.

Response: Optional. Fixed length of 3 bytes.

A numeric code indicating the currency of the transaction, as received from the POS device, according to the ISO 4217 standard, ***Codes for the Representation of Currencies and Funds***.

SFID J — Cardholder Certificate Serial Number

Request: Optional. Fixed length of 32 bytes.

Response: Not applicable.

The cardholder certificate serial number for secure electronic commerce.

SFID K — Merchant Certificate Serial Number

Request: Optional. Fixed length of 32 bytes.

Response: Not applicable.

The merchant certificate serial number for secure electronic commerce.

SFID L — XID/TRANS STAIN

Request: Optional. Fixed length of 80 bytes. The first 40 bytes contain the transaction identifier. The second 40 bytes contain the transaction stain.

Response: Not applicable.

The transaction identifier and transactions stain. The transaction identifier (XID) is 40 bytes long. The transaction stain is a hash value calculated by applying a secure hash algorithm to the XID and CardSecret (a secret SET-defined value known only to the cardholder and the issuer of the cardholder certificate). It is 40 bytes long.

SFID N — Message Reason Code

Request: Optional. Fixed length of 4 bytes.

Response: Not applicable.

The message reason code specifies why a transaction is to be authorized online (rather than being completed locally), or why a transaction has been completed locally (rather than being authorized online). Values are defined in the ISO 8583 (1993) standard.

In an online request message (message subtype O), the message reason code contains one of the values in the table below. When more than one message reason code condition applies to a transaction, the applicable message reason code with the highest priority is used.

Value	Priority	Description	Example Conditions
1500	9	ICC application, common data file (CDF) unable to process	Only used by integrated ICC/MSR terminals where the terminal has possession of the card and this condition can be accurately identified.
1501	10	ICC application, application data file (ADF) unable to process	Only used by integrated ICC/MSR terminals where the terminal has possession of the card and this condition can be accurately identified.
1502	11	ICC random selection	Not used.
1503	6	Terminal random selection	The "Transaction selected randomly for online processing" bit is set in byte 4, bit 5 of the Terminal Verification Results (TVR) in the EMV Request Data (SFID O).
1504	1	Terminal not able to process ICC	Fallback for ICC.
1505	2	Online forced by ICC	ICC forced online, no bits set in TVR.
1506	3	Online forced by card acceptor	Card used twice; used the maximum times per day; one in *n* authorization; PAN key entry authorization; exclusion band checks; prevalid card.
1507	12	Online forced by card accepting device (CAD) to be updated	Not used.
1508	8	Online forced by terminal	The TVR indicates that online processing is required.
1509	4	Online forced by card issuer	Expired card; new card; service code; hot card.
1510	7	Over floor limit	Transaction amount above floor limit.
1511	5	Merchant suspicious	Force authorization ("no" at signature check); card returns an inappropriate cryptogram.

In a force-post message (message subtype F) or store-and-forward message (message subtype S) that the terminal has previously attempted to send to the host as an online request message (message subtype O), the message reason code contains the same value as in the online request message.

In a force-post message (message subtype F) or store-and-forward message (message subtype S) that the terminal has not previously attempted to send to the host as an online request message (message subtype O), the message reason code contains one of the values in the table below. When more than one message reason code condition applies to a transaction, the applicable message reason code with the highest priority is used.

Value	Priority	Description	Example Conditions
1004	1	Terminal processed	Forced data capture (e.g., local transaction store is full).
1005	2	ICC processed	ICC authorized.
1006	3	Under floor limit	Amount is less than the terminal's floor limit in a non-ICC transaction.
1007	4	Stand-in processing at the acquirer's option	A preauthorization completion transaction accepted by the terminal.

Acquirers can use the message reason code to determine the appropriate EMV Authorization Response Code (ARC) for transactions authorized at a terminal. If an approved advice message is received with a message reason code of 10xx, then the ARC is Y1 (offline approved). If an approved advice message is received with a message reason code of 15xx, then the ARC is Y3 (unable to go online, offline approved).

SFID O — EMV Request Data

Request: Optional. Variable length of 121–136 bytes, depending on the Issuer Application Data field definition used.

Response: Not applicable.

The EMV Request Data field contains the thirteen minimum request data elements, as defined by EMV. This subFID is required for all EMV transaction requests. The format of this subFID depends on the value in the Smart Card Scheme field.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

For terminals supporting the EMV version 1 scheme, the format is as follows:

Smart Card Scheme	PIC 9(2)
Cryptographic Information Data	PIC 9(2)
Terminal Country Code	PIC 9(3)
EMV Date	PIC 9(6)
Application Cryptogram (AC)	PIC X(16)
Application Interchange Profile (AIP)	PIC X(4)
Application Transaction Counter (ATC)	PIC X(4)
Unpredictable Number	PIC X(8)
Terminal Verification Result (TVR)	PIC X(10)
Cryptogram Transaction Type	PIC 9(2)
Issuer Application Data	PIC X(64) variable data

For terminals supporting the EMV version 2 scheme, the format is as follows:

Smart Card Scheme	PIC 9(2)
Cryptographic Information Data	PIC 9(2)
Terminal Country Code	PIC 9(3)
EMV Date	PIC 9(6)
Application Cryptogram (AC)	PIC X(16)
Application Interchange Profile (AIP)	PIC X(4)
Application Transaction Counter (ATC)	PIC X(4)
Unpredictable Number	PIC X(8)
Terminal Verification Result (TVR)	PIC X(10)
Cryptogram Transaction Type	PIC 9(2)
Cryptogram Currency Code	PIC X(3)
Cryptogram Amount	PIC X(12)
Issuer Application Data	PIC X(64) variable data

These fields are further described below:

Position	Length	Description
01–02	**2**	**Smart Card Scheme** A code indicating the smart card scheme used for this transaction. Valid values are as follows: 00 = EMV version 1 01 = EMV version 2

Position	Length	Description
03–04	**2**	**Cryptographic Information Data** Hexadecimal characters (0–F) representing eight bits of cryptographic information data. The first hexadecimal character represents bits 0–3, and the second hexadecimal character represents bits 4–7. The host converts the hexadecimal characters received in the request message to binary data for storage. **EMV Tag:** 9F27.
05–07	**3**	**Terminal Country Code** A three-digit code indicating the country where this terminal is located, according to the ISO 3166 standard, ***Codes for the Representation of Names of Countries***. **EMV Tag:** 9F1A.
08–13	**6**	**EMV Date** The local date (in YYMMDD format) that the transaction was authorized. **EMV Tag:** 9A.
14–29	**16**	**Application Cryptogram (AC)** The cryptogram returned by the ICC in response to the GENERATE AC command. The field contains the Transaction Certificate generated by the Europay, MasterCard, and Visa (EMV) card for an offline EMV transaction. **EMV Tag:** 9F26.

Position	Length	Description
30–33	**4**	**Application Interchange Profile (AIP)** Hexadecimal characters (0–F) representing 16 bits of Application Interchange Profile data. The first hexadecimal character represents bits 0–3 of byte 1, the second hexadecimal character represents bits 4–7 of byte 1, the third hexadecimal character represents bits 0–3 of byte 2, and so on. The host converts the hexadecimal characters received in the request message to binary data for storage. **EMV Tag:** 82.
34–37	**4**	**Application Transaction Counter (ATC)** A count of the number of transactions performed with this EMV card. The application on the chip maintains and increments the application transaction counter. **EMV Tag:** 9F36.
38–45	**8**	**Unpredictable Number** An unpredictable number, in hexadecimal format, used to provide variability and uniqueness to the generation of a cryptogram. **EMV Tag:** 9F37.
46–55	**10**	**Terminal Verification Results (TVR)** Multiple bit values stored as hexadecimal characters (0–F). The first hexadecimal character represents bits 0–3 of byte 1, the second hexadecimal character represents bits 4–7 of byte 1, the third hexadecimal character represents bits 0–3 of byte 2, and so on. The host converts the hexadecimal characters received in the request message to binary data for storage. **EMV Tag:** 95.

Position	Length	Description
56–57	**2**	**Cryptogram Transaction Type** A code indicating the type of financial transaction, represented by the first two digits of the processing code from the 1987 ISO 8583 standard, ***Bank Card Originated Messages—Interchange Message Specifications—Content for Financial Transactions***. **EMV Tag:** 9C.
58–60	**3**	**Cryptogram Currency Code** A code indicating the currency used for the transaction amount in the Cryptogram Amount field. **Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme. **EMV Tag:** 5F2A.
61–72	**12**	**Cryptogram Amount** The transaction amount used to generate the value in the Application Cryptogram (AC) field. **Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme. **EMV Tag:** 9F02.
58–121 or 73–136	**64**	**Issuer Application Data** Contains proprietary issuer application data for transmission to the issuer in an online transaction. The host currently supports three definitions for issuer application data. For more information on these fields, refer to the individual card scheme documentation. **Note:** The position of this field depends on which smart card scheme is used. **EMV Tag:** 9F10.

Position	Length	Description
58–121	**64**	**Visa/UKIS Definition** The following fields contain the Visa/UKIS definition of the Issuer Application Data field.
58–59	**2**	**Length**
60–61	**2**	**Derivation Key Index**
62–63	**2**	**Cryptogram Version Number**
64–71	**8**	**Card Verification Results**
72–121	**50**	**Issuer Discretionary Data**
58–121	**64**	**MasterCard/Europay M/CHIP 2.1 Definition** The following fields contain MasterCard/Europay (MCPA) M/CHIP 2.1 definition of the Issuer Application Data field.
58–59	**2**	**Derivation Key Index**
60–61	**2**	**Cryptogram Version Number**
62–69	**8**	**Card Verification Results**
70–73	**4**	**Dynamic Authentication Code**
74–121	**48**	**Issuer Discretionary Data**
73–136	**64**	**MasterCard/Europay M/CHIP 4 Definition** The following fields contain the MasterCard/Europay M/CHIP 4 definition of the Issuer Application Data field.
73–74	**2**	**Derivation Key Index**
75–76	**2**	**Cryptogram Version Number**
77–84	**12**	**Card Verification Results**
85–88	**4**	**Dynamic Authentication Code**
89–104	**16**	**Counters**
105–136	**28**	**Issuer Discretionary Data**

Position	Length	Description
73–136	**64**	**EMV CCD Definition** The following fields contain the EMV CCD definition of the Issuer Application Data field.
73–74	**2**	**Length**
75–76	**2**	**Common Core ID**
77–78	**2**	**Derivation Key Index**
79–88	**10**	**Card Verification Results**
89–104	**16**	**Counters**
105–106	**2**	**Issuer Discretionary Data Length**
107–136	**30**	**Issuer Discretionary Data**

Note: EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes only.

SFID P — EMV Additional Request Data

Request: Optional. Variable length of 64 bytes. Currently only 48 bytes are used.

Response: Not applicable.

The EMV Additional Request Data field contains additional EMV transaction data. This subFID is optional for all EMV transaction requests. The format of this subFID depends on the value in the Smart Card Scheme field.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

For terminals supporting the EMV version 1 scheme, the format is as follows:

Smart Card Scheme	PIC 9(2)
Application PAN Sequence Number	PIC 9(2)
EMV Terminal Type	PIC 9(2)

For terminals supporting the EMV version 2 scheme, the format is as follows:

Smart Card Scheme	PIC 9(2)
Application PAN Sequence Number	PIC 9(2)
EMV Terminal Type	PIC 9(2)
Cardholder Verification (CVM) Results	PIC X(6)
Application Version Number	PIC X(4)
Dedicated File Name	PIC X(32) variable data

For terminals supporting the EMV version 3 scheme, the format is as follows:

Smart Card Scheme	PIC 9(2)
Application PAN Sequence Number	PIC 9(2)
EMV Terminal Type	PIC 9(2)
Cardholder Verification (CVM) Results	PIC X(6)
Application Version Number	PIC X(4)
EMV Terminal Capabilities	PIC X(6)
Dedicated File Name	PIC X(32) variable data

If an additional EMV transaction data element is not present, the corresponding field in this subFID is space-filled.

These fields are further described below:

Position	Length	Description
01–02	**2**	**Smart Card Scheme** A code indicating the smart card scheme used for this transaction. Valid values are as follows: 00 = EMV version 1 01 = EMV version 2 02 = EMV version 3

Position	Length	Description
03–04	**2**	**Application PAN Sequence Number** The application PAN sequence number. This field identifies and differentiates cards with the same PAN. **EMV Tag:** 5F34.
05–06	**2**	**EMV Terminal Type** The EMV terminal type, indicating the environment of the terminal, its communications capability, and its operational control. **EMV Tag:** 9F35.
07–12	**6**	**Cardholder Verification (CVM) Results** Hexadecimal characters (0–F) representing multiple values. The first hexadecimal character represents bits 0–3 of byte 1, the second hexadecimal character represents bits 4–7 of byte 1, the third hexadecimal character represents bits 0–3 of byte 2, and so on. The host converts the hexadecimal characters received in the request message to binary data for storage. **Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme. **EMV Tag:** 9F34.
13–16	**4**	**Application Version Number** The application version number assigned by the payment system for the application. **Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme. **EMV Tag:** 9F09.

Position	Length	Description
17–22	**6**	**EMV Terminal Capabilities***
		Indicates the card data input, CVM, and security capabilities of the terminal. Refer to the EMV-TERM-CAP field in the DDLGDEFS file on the BA60DEFS subvolume for additional information about terminal capabilities.
		* This field is used for the version 3 scheme only. For the version 2 scheme, the Dedicated File Name field starts at position 17.
23–54	**32**	**Dedicated File Name**
		The name of the dedicated file (as described in ISO/IEC 7816-4) or application identifier (as described in ISO/IEC 7816-5).
		Note: This field exists only for terminals supporting the EMV version 2 smart card scheme.
		EMV Tag: 84.

Note: EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes only.

SFID Q — EMV Response Data

Request: Not applicable.

Response: Optional. Variable length of 64 bytes. Currently only 36 bytes are used.

The EMV Response Data field contains the response cryptogram and the response code used to generate the response cryptogram. This subFID is required for EMV transaction response messages for which an ARPC has been generated. The format of this subFID depends on the value in the Smart Card Scheme field.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

For terminals supporting the EMV version 1 scheme, the format is as follows:

Smart Card Scheme	PIC 9(2)
Issuer Authentication Data	PIC X(32) variable length

For terminals supporting the EMV version 2 scheme, the format is as follows:

Smart Card Scheme	PIC 9(2)
Authorization Response Code	PIC X(2)
Issuer Authentication Data	PIC X(32) variable length

These fields are further described below:

Position	Length	Description
01–02	**2**	**Smart Card Scheme** A code indicating the smart card scheme used for this transaction. Valid values are as follows: 00 = EMV version 1 01 = EMV version 2

Position	Length	Description
03–04	**2**	**Authorization Response Code** The EMV authorization response code used with the EMV version 2 card scheme. Valid values are based on host response codes, as follows: 00 = Approve (host response codes 000–049) 01 = Refer (host response codes 100–149) 04 = Capture (host response codes 900–949) 05 = Decline (all other host response codes) **Note:** This field exists only for terminals supporting the EMV version 2 smart card scheme. **EMV Tag:** 8A.
03–34 or 05–36	**32**	**Issuer Authentication Data** Contains proprietary issuer authentication data for transmission to the card in an online transaction. For more information on the format of this field, refer to the individual card scheme documentation. **EMV Tag:** 91.

Note: EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes only.

SFID R — EMV Reversal Data/EMV Additional Response Data

Request: Not present in an original request but might be present in a reversal. Variable length of 82 bytes.

Response: Optional. Variable length of 258 bytes.

The EMV Reversal Data/EMV Additional Response Data field contains EMV script data returned in an EMV transaction response from the card issuer. It also can be used in a subsequent reversal message to deliver EMV script results, if required. This subfield is required in EMV transaction response messages only if a script is present for script processing.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

The format is as follows:

Smart Card Scheme	PIC 9(2)
Issuer Script Data	PIC X(256) variable length

These fields are further described below:

Position	Length	Description
01–02	**2**	**Smart Card Scheme** A code indicating the smart card scheme used for this transaction. Valid values are as follows: 00 = EMV version 1 01 = EMV version 2
03–258	**256**	**Issuer Script Data** The issuer script template, in hexadecimal format, to be sent to the card for processing by the card application. **EMV Tag:** 71 or 72.

Note: EMV tags are hexadecimal identifiers for data elements in EMV specifications. They are provided in this SFID description for reference purposes only.

The format for returning EMV script results in a reversal message is as follows:

EMV Scheme	PIC X(2)
EMV Issuer Script Results	Occurs one to eight times
Issuer Script Processing Result	PIC X
Issuer Script Sequence Number	PIC X
Issuer Script Identifier	PIC X(8)

These fields are further described below:

Position	Length	Description
01–02	**2**	**Smart Card Scheme** A code indicating the smart card scheme used for this transaction. Valid values are as follows: 00 = EMV version 1
03–82	**10–80**	**EMV Issuer Script Results** Indicates the result of terminal script processing. EMV does not define a tag for this data element.
03	**1**	**Issuer Script Processing Result** A code indicating the result of the script processing. Valid values are as follows: 0 = Script not performed 1 = Script processing failed 2 = Script processing successful 9 = Script processing unknown
04	**1**	**Issuer Script Sequence Number** A code indicating the details of the script sequence used in the processing. Valid values are as follows: 0 = Script sequence not specified, script not performed, or all commands successful. 1 = "E" sequence number from 1–14 for failed command. F = Sequence number of 15 or over for failed command.
05–12	**8**	**Issuer Script Identifier** The issuer script identifier, if available. If the issuer script identifier is not available, this field is zero filled.

SFID S — Stored Value Data

Request: Optional. Fixed length of 63 bytes.

Response: Optional. Fixed length of 63 bytes.

The Stored Value Data field contains stored value card information. This subFID is required for additional card activation transaction requests and all stored value transaction responses. The format is as follows:

Balance as Cash Flag	PIC X
Card Balance	PIC 9(16)v99
Expiration Date	PIC X(4)
Additional Track2	PIC X(40)

These fields are further described below:

Position	Length	Description
01	**1**	**Balance as Cash Flag** A flag indicating whether the remaining balance on the stored value account can be given as cash. This flag is provided in response messages only. Valid values are as follows: 0 = Do not give the remaining account balance as cash. 1 = Give the remaining account balance as cash.
02–19	**18**	**Card Balance** The remaining balance on the stored value account. This amount is returned in stored value response messages only.
20–23	**4**	**Expiration Date** The expiration date of this stored value account in YYMM format. This amount is returned in stored value response messages only.
24–63	**40**	**Additional Card Track 2 Data** The Track 2 data from the additional card. This data is present only in additional card activation transaction request messages from the terminal.

SFID T — Key Serial Number and Descriptor

Request: Optional. Variable length of 23 bytes.

Response: Not applicable.

The key serial number (KSN) and key descriptor are used with derived unique key per transaction (DUKPT) processing. The format is as follows:

Key Serial Number	PIC X(20)
Key Serial Number Descriptor	PIC X(3)

These fields are further described below:

Position	Length	Description
01–20	**20**	**Key Serial Number** The key serial number (KSN) associated with the transaction. The layout of this field is as follows: *FFFFkkkkkkkkkkssss* where, F = "F" pad character (4 characters) k = Static Key Serial Number data (11 characters) (static for all transactions) s = Transaction counter (5 characters) (incremented for each transaction) An example of Key Serial Numbers for two consecutive transactions might be as follows: FFFF9876543210E00001 FFFF9876543210E00002 In this example, if the same PAN and PIN were used for both of transactions, the DUKPT-encrypted PIN block would look completely different due to the transaction counter change.
21–23	**3**	**Key Serial Number Descriptor** The key serial number descriptor from the terminal. This field is required only if a Thales e-Security (Racal) security module is being used by the host.

SFID U — Transaction Subtype Data

Request: Optional. Fixed length of 16 bytes.

Response: Not applicable.

Transaction subtype data enables an institution to distinguish between types of transactions that possess identical transaction codes. For example, transaction subtypes enable an institution to configure multiple rate structures for a single transaction code. The format is as follows:

Transaction Subtype	PIC X(4)
Acquirer Processing Code	PIC X(6)
Issuer Processing Code	PIC X(6)

These fields are further described below:

Position	Length	Description
01–04	**4**	**Transaction Subtype** A subtype identifier to further describe this transaction. All alphanumeric characters are valid in this field. Values P000 through RZZZ are reserved for host-defined transaction subtypes. Values 0000 through OZZZ and S000 through ZZZZ are reserved by ACI. Valid subtype values are as follows: B000 Payment from Third Party B001 Payment to Third Party BBT0 BCGI Top-Up C000 Account Funding Transaction C001 Healthcare/Transit Auto-Substantiation C002 Healthcare Eligibility Inquiry C003 Dormancy Transaction C004 Escheatment Transaction C005 Payment Transaction C006 Original Credit Transaction C007 Loyalty/Sweepstakes/Extras Transactions C008 Quasi-cash Transaction CI00 Canadian Idebit

Position	Length	Description
05–10	**6**	**Acquirer Processing Code** The acquirer's external processing code. The first two characters of the processing code indicate the type of transaction, the next two characters specify the *from* account for the transaction, and the last two characters specify the *to* account for the transaction.
11–16	**6**	**Issuer Processing Code** The issuer's external processing code. The first two characters of the processing code indicate the type of transaction, the next two characters specify the *from* account for the transaction, and the last two characters specify the *to* account for the transaction.

SFID V — Authentication Collection Indicator

Request: Optional. Fixed length of 1 byte.

Response: Optional. Fixed length of 1 byte.

This field identifies whether customer authentication data is supported and whether the data is available. Valid values are as follows:

0 = Universal Cardholder Authentication field (UCAF) data collection is not supported at the merchant's Web site.
1 = UCAF data collection is supported by the merchant, but UCAF data was not populated.
2 = UCAF data collection is supported by the merchant, and UCAF data was populated.
3 = UCAF data collection is supported by the merchant, and UCAF data was populated by a static authentication value.

SFID W — CAVV/AAV Result Code

Request: Not applicable.

Response: Optional. Fixed length of 1 byte.

The CAVV/AAV result code indicates the result of the CAVV (VISA method) or AAV (MasterCard method) validation. Valid values are as follows:

0 = Not validated due to erroneous data
1 = Failed validation - authentication
2 = Passed validation - authentication
3 = CAVV passed validation - attempt. Authentication was attempted at the issuer's ACS but did not complete successfully.
4 = CAVV failed validation - attempt at issuer's ACS.
5 = The acquirer is participating in authentication, but the issuer is not participating
6 = CAVV not validated; issuer BIN not participating in CAVV validation.
7 = CAVV failed validation - attempt.
8 = CAVV passed validation - attempt. Authentication was attempted at the interchange's ACS but did not complete successfully.
9 = CAVV failed validation - attempt at interchange during stand-in.
A = CAVV passed validation - attempt at interchange during stand-in.
B = CAVV passed validation - information only, no liability shift.
C = CAVV not validated, attempt. This issuer did not return the results code in the authorization response.
D = CAVV was not validated - authentication. The issuer failed to return the result value.
W = CAVV/AAV validation could not be performed (no EAF).
X = CAVV/AAV validation could not be performed due to system error, or failure prevented authentication (error accessing EAF).
Y = The acquirer is participating in authentication but the issuer is not participating.
Z = Duplicate CAVV/AAV.

SFID X — Point of Service Data

Request: Optional. Fixed length of 6 bytes.

Response: Not applicable.

Point of service data includes a group of flags that indicate card, cardholder, and transaction status associated with a point-of-sale transaction. The format is as follows:

Cardholder Present Indicator	PIC X
Card Present Indicator	PIC X
Transaction Status Indicator	PIC X
Transaction Security Indicator	PIC X
Cardholder Activated Terminal Indicator	PIC X
Cardholder ID Method	PIC X

These fields are further described below:

Position	Length	Description
01	**1**	**Cardholder Present Indicator** A code indicating whether the cardholder is present at the POS terminal. Valid values are as follows: 0 = The cardholder is present 1 = The cardholder is not present—unspecified reason 2 = The cardholder is not present—mail or fax order 3 = The cardholder is not present—telephone or ARU order 4 = The cardholder not present—standing order or recurring transaction 5 = The cardholder is not present—electronic order (home personal computer or Internet)
02	**1**	**Card Present Indicator** A code indicating whether the card is present at the POS terminal. Valid values are as follows: 0 = The card is present 1 = The card is not present

Position	Length	Description
03	**1**	**Transaction Status Indicator** A code indicating the purpose or status of the request. Valid values are as follows: 0 = Normal request 1 = Merchant authorization 4 = Preauthorized request 5 = Stand-in 6 = Address verification request 7 = Cash back 8 = Downtime submission request
04	**1**	**Transaction Security Indicator** A code indicating the card acceptor's security level. Valid values are as follows: 0 = No security concern 1 = Suspected fraud (merchant suspicious—code 10) 2 = Identification verified
05	**1**	**Cardholder Activated Terminal Indicator** A code indicating whether the cardholder activated the terminal with a card, and if so, the level of security. Valid values are as follows: 0 = The transaction is not a CAT transaction 1 = Automated dispensing machine with PIN—level 1 security 2 = Self-service terminal or contactless/proximity terminal—level 2 security 3 = Limited amount terminal—level 3 security 4 = In-flight commerce—level 4 security 5 = Script device 6 = Electronic commerce 7 = Radio frequency device

Position	Length	Description
06	1	**Cardholder ID Method** A code indicating how the cardholder was verified at the point-of-service. Valid values are as follows: 0 = Unknown (default) 1 = Signature 2 = PIN 3 = None (Cardholder Present) 4 = None (Cardholder Not Present) 5 = Authentication Value 6 = Electronic Signature Analysis 7 = Biometrics 8 = Biographics 9 = Other

SFID Y — Authentication Data

Request: Optional. Variable length of 2–202 bytes.

Response: Optional. Variable length of 2–202 bytes.

Authentication Data standardizes the transport cardholder authentication data for e-commerce transactions. The format is as follows:

Authentication Indicator Flag	PIC X(2)
Authentication Indicator Data	PIC X(200), variable length

These fields are further described below:

Position	Length	Description
01–02	02	**Authentication Indicator Flag** The authentication indicator flag. The only valid value for this field is 01 (Universal Cardholder Authentication Field).

Position	Length	Description
03–202	**200**	**Authentication Indicator Data** The generic data. For a MasterCard transaction using the Secure Payment Application Universal Cardholder Authentication field (SPA UCAF) method, this field contains the Accountholder Authentication Value (AAV).

SFID Z — Card Verification Flag 2

Request: Optional. Fixed length of 1 byte.

Response: Optional. Fixed length of 1 byte.

Card Verification Flag 2 indicates whether the card involved in a card-read transaction has already been verified using the CVV2/CVD2/CSC. Valid values are as follows:

0 = Card verification was not performed because the transaction was denied before card verification processing started.

C = Card verification was performed and the card verification digits (CVD) were invalid. The situation was noted and the transaction processing continued.

D = Card verification was performed and the CVD was invalid. The transaction was denied.

N or ~~b~~ = Card verification was not attempted or a security device error occurred (where ~~b~~ indicates a blank space).

O = Card verification was not performed, a CVD value was not on the card. Not all cards have a CVD value encoded. The card expiration date must be equal to or greater than an expiration date defined on the CPF to ensure that the CVD field has been encoded. If the card expiration date is equal to or greater than the CPF date, the CVD checks are performed.

P = Card verification was not performed. Either the merchant ignored the CVD value on purpose or the user falsely indicated no CVD was on the card.

R = Card verification was performed and the CVD was invalid. The situation was noted and the transaction should be referred.

S = CVV2 should be on the card but the merchant indicates that it is not.

U = Issuer has not certified or has not provided the encryption keys to the interchange.

Y = Card verification was performed and the CVD was valid.

SFID b — Electronic Check Conversion Data

Request: Optional. Fixed length of 39 bytes.

Response: Not applicable

Electronic check conversion information used with check guarantee or check verification transactions for electronic check authorization. The first 38 bytes of this field contains magnetic ink character recognition (MICR) data containing the institution routing number, account number, and check number of the check to be converted. The next byte contains a flag indicating the type of conversion to perform. The format is as follows:

Institution Routing Number	PIC X(11)
Account Number	PIC X(19)
Check Number	PIC X(8)
Conversion Flag	PIC X

These fields are further described below:

Position	Length	Description
01–11	**11**	**Institution Routing Number** The institution routing number from the MICR data.
12–30	**19**	**Account Number** The account number from the MICR data.
31–38	**8**	**Check Number** The check number from the MICR data.

Position	Length	Description
39	1	**Conversion Flag** A flag indicating the type of electronic conversion to perform. Valid values are as follows: 0 = No conversion. 1 = Perform a check verification or check guarantee transaction with conversion. 2 = Conversion only transaction.

SFID c — MICR Data

Request: Optional. Fixed length of 64 bytes.

Response: Not applicable

The magnetic ink character recognition (MICR) data for the check in Raw Toad format (i.e., as read by the MICR reader). If this subFID is present in the message, the MICR data in the Electronic Check Conversion Data field (subFID b) is parsed from this data. If this subFID is not present in the message, the MICR data in the Electronic Check Conversion Data field (subFID b) is manually entered.

This field is not used by the host for processing. Only SFID b is used by the host and must be present. SFID c is present only if an interchange requires the full MICR in the Raw Toad format.

SFID d — Electronic Check Callback Information

Request: Optional. Fixed length of 115 bytes.

Response: Optional. Fixed length of 115 bytes.

The callback information provided on a customer receipt. A request message contains this subFID only when the customer phone number or process control number are received from the device. A response message contains this subFID only when the external message received from the authorizer contains the non-bank authorizer information.

The format is as follows:

Customer Phone Number	PIC X(20)
Process Control Number	PIC X(6)
Non-bank Authorizer's Name	PIC X(25)
Non-bank Authorizer's Street	PIC X(20)
Non-bank Authorizer's City	PIC X(13)
Non-bank Authorizer's State	PIC X(2)
Non-bank Authorizer's Postal Code	PIC X(9)
Non-bank Authorizer's Phone Number	PIC X(20)

These fields are further described below:

Position	Length	Description
01–20	**20**	**Customer Phone Number** The customer's telephone number, if available. This field is filled in by the host, and is blank-filled in a response when the information has not been provided in the request.
21–26	**6**	**Process Control Number** The process control number from the check. This field is filled in by the host, and is blank-filled in a response when the information has not been provided in the request.
27–51	**25**	**Non-bank Authorizer's Name** The name of the authorizer, provided in the response by the authorizer.
52–71	**20**	**Non-bank Authorizer's Street** The street address of the authorizer, provided in the response by the authorizer.
72–84	**13**	**Non-bank Authorizer's City** The city of the authorizer, provided in the response by the authorizer.
85–86	**2**	**Non-bank Authorizer's State** The state of the authorizer, provided in the response by the authorizer.

Position	Length	Description
87–95	**9**	**Non-bank Authorizer's Postal Code** The postal code of the authorizer, provided in the response by the authorizer.
96–115	**20**	**Non-bank Authorizer's Phone Number** The phone number of the authorizer, provided in the response by the authorizer.

SFID e — Interchange Compliance Data

Request: Not applicable.

Response: Optional. Fixed length of 21 bytes.

This field contains interchange compliance data associated with a MasterCard transaction. The format is as follows:

Trace ID	PIC X(15)
Valid Code	PIC X(4)
Monitoring Status	PIC X
Error Indicator	PIC X

These fields are further described below:

Position	Length	Description
01–15	**15**	**Trace ID** The code assigned by the interchange to a transaction that has met the required compliance edits. A combination of the Network ID, Reference number, and date is filled into this field, depending on the interchange.
16–19	**4**	**Valid Code** The code assigned by the interchange to a transaction that has met the required compliance edits and has been approved by the issuer.

Position	Length	Description
20	**1**	**Monitoring Status** A code returned from the interchange indicating whether MasterCard changed the Point of Service Entry Mode from 90 to 02. A value of Y indicates that the status is being monitored.
21	**1**	**Error Indicator** A code returned from the interchange indicating an error condition that may have occurred. Valid values are as follows: ␣ = No error occurred (where ␣ indicates a blank space) A = Track 1 or Track 2 data not present in message B = Track 1 and Track 2 data present in message C = PAN not equal in PAN data D = Expiration date not equal in PAN data E = Card type invalid in track data F = Field separator(s) invalid in track data G = A field within the track data exceeds the maximum length H = Transaction category code is T I = POS customer presence indicator is 1 J = POS card presence indicator is 1

SFID f — Response Source or Reason Code

Request: Not applicable.

Response: Optional. Fixed length of 1 byte.

This subfield contains the response source or reason code set by an interchange to provide additional information regarding a response. Valid values are as follows:

1 = Request timed out at interchange
2 = Transaction amount below issuer limit
3 = Issuer is in suppress inquiries mode
4 = Issuer is not available for processing

5 = Response provided by issuer
7 = Reversal advice provided by interchange to identify a potential duplicate transaction
8 = Reversal advice provided by interchange to identify a probable duplicate authorization
A = Third party agent

The processing of EMV transactions assumes that all real-time authorization requests are sent to the card issuer for approval. In some instances, however, an intermediate system (for example, an acquirer or card acceptor) stands in for the issuer and authorizes the transaction. In this situation, the response message does not contain an Authorization Response Cryptogram (ARPC), that is, the Issuer Authentication Data (subFID 6Q) is not present, and the integrated circuit card may subsequently decline the transaction if the response purports to come from the issuer.

To avoid this situation, the intermediate system can use this subfield to indicate that a system other than the card issuer is the authorizing entity. If a value of 1 or 4 is returned in this subfield, then the transaction could not be sent to the issuer for authorization. If the Issuer Authentication Data (subFID 6Q) is not present in the response message, then the terminal should respond to the integrated circuit card with an EMV response code of Y3 (unable to go online, offline approved) or Z3 (unable to go online, offline declined). If a value of 2 or A is returned in this subfield, then the transaction was authorized on behalf of the issuer. If the Issuer Authentication Data (subFID 6Q) is not present in the response message, then the terminal should respond to the integrated circuit card with an EMV response code of Y1 (offline approved) or Z1 (offline declined).

SFID g — POS Merchant Data

Request: Optional. Fixed length of 4 bytes.

Response: Not applicable.

This subfield contains additional POS merchant data. The format is as follows:

E-Commerce Goods Indicator	PIC X
Existing Debt Indicator	PIC X
Deferred Billing Indicator	PIC X
Relationship Participant Indicator	PIC X

These fields are further described below:

Position	Length	Description
01	**1**	**E-Commerce Goods Indicator** A code indicating the type of merchandise being sold. This code is passed from the acquiring terminal and can be specific to a transaction or merchant. Valid values are as follows: D = Digital P = Physical goods S = Services
02	**1**	**Existing Debt Indicator** A code indicating whether a credit card is used to pay for an existing debt. If a credit card was not used to pay for an existing debt, this field is blank. This field is passed from the acquiring terminal. The only valid value is 9 (payment on existing debt).
03	**1**	**Deferred Billing Indicator** A code indicating whether a purchase is made with the payment deferred until a later date. This code is passed from the acquiring terminal. Valid values are as follows: 0 = Deferred billing is not provided. 1 = Deferred billing is used at the point of service.
04	**1**	**Relationship Participant Indicator** A code indicating whether the merchant or acquirer has a special relationship with the cardholder. This code is passed from the acquiring terminal. Valid values are as follows: 0 = Not a relationship participant or relationship not provided. 1 = Relationship participant.

SFID h — System Trace Audit Number (STAN)

Request: Not applicable

Response: Optional. Fixed length of 6 bytes.

This subfield contains a number assigned by a transaction originator to uniquely identify a transaction. The trace number remains unchanged for all messages within the transaction.

SFID i — Retrieval Reference Number

Request: Not applicable

Response: Optional. Fixed length of 12 bytes.

This subfield contains a reference number supplied by the system that retains the original source information.

SFID j — Debit Network/Sharing Group ID

Request: Not applicable

Response: Optional. Fixed length of 4 bytes.

This subfield is a network identifier for the message transmission and defines the program rules that apply to the transaction. It is usually set by the interchange network that acquires the transaction.

SFID k — Card Level Results

Request: Not applicable.

Response: Optional. Fixed length of 2 bytes.

This subfield contains a code indicating the participation program for the card involved in the transaction. The first byte has one of the following values:

A	=	Visa Traditional Non-Rewards – Identifies any consumer credit card that does not offer rewards or meet the Visa Traditional Rewards product requirements and program standards. Default.
AX	=	American Express
B	=	Visa Traditional Rewards / MasterCard Enhanced (Consumer)
C	=	Visa Signature / MasterCard Consumer World
D	=	Visa Signature Preferred / MasterCard Consumer World Elite
DI	=	Discover
G	=	Visa Business/MasterCard Business World
G1	=	Visa Signature Business
G2	=	Visa Business Check Card
H	=	Visa Check Card/MasterCard Business World Elite
I	=	Visa Commerce/MasterCard Corporate World
J	=	MasterCard Corporate World Elite
J1	=	Visa General Prepaid
J2	=	Visa Prepaid Gift
J3	=	Visa Prepaid Healthcare
J4	=	Visa Prepaid Commercial
K	=	Visa Corporate
K1	=	Visa GSA Corporate T & E
M	=	MasterCard/Euro Card and Diners
Q	=	Private Label
Q1	=	Private Label Prepaid
R	=	Proprietary
S	=	Visa Purchasing
S1	=	Visa Purchasing with Fleet
S2	=	Visa GSA Purchasing
S3	=	Visa GSA Purchasing with Fleet
U	=	Visa TravelMoney
Z	=	Does not participate (MasterCard default value)

The second byte is a blank, A-Z or 0-9. All are reserved values.

SFID I — Healthcare/Transit Data

Request: Optional. Variable length of 20-120 bytes.

Response: Optional. Variable length of 20-120 bytes.

This subfield contains information associated with healthcare/transit auto-substantiation transactions and healthcare eligibility inquiry transactions. The format is as follows:

Additional Amount	occurs 1 to 6 times
Account Type	PIC X(2)
Amount Type	PIC X(2)
Currency Code	PIC X(3)
Amount Sign	PIC X
Amount	PIC X(12)

For healthcare/transit auto-substantiation transactions, only entry 1 of the Additional Amount table is used.

For healthcare eligibility inquiry transactions, entries 1 through 6 of the Additional Amount table are used.

These fields are further described below.

Position	Length	Description
01-120	20-120	**Additional Amount (occurs 1 to 6 times)** Additional amount information.
01-02	2	**Account Type** The type of account being used. A value of 00 indicates a non-specified type will be used for healthcare/transit auto-substantiation transactions and healthcare eligibility inquiry transactions.

03-04	2	**Amount Type** The type of payment amount. Valid values are as follows: 3S = Amount co-payment 4S = Amount healthcare 4T = Amount transit 4U = Amount prescription/Rx 4V = Amount vision/optical 4W = Amount clinic/other qualified medical 4X = Amount dental 57 = Original amount Merchants insert the total amount of qualified healthcare products as amount type 4S. This includes over-the-counter (OTC) products and other applicable types of qualified medical expenses (non-OTC products), such as prescriptions, vision/optical, and clinic products. Some health plans only cover specific types of expenditures, such as prescription or vision. Thus, they can optionally include additional information in amount type fields for prescription/Rx (4U), vision/optical services (4V), clinic/other qualified medical services (4W), and dental (4X).
05-07	3	**Currency Code** The standard ISO Currency Code of the amount.
08	1	**Amount Sign** A code indicating whether the amount is positive or negative. Valid values are as follows: C = Credit, positive balance D = Debit, negative balance
09-20	12	**Amount** The amount specified by the Amount Type field (right-justified and zero-filled on the left).

When this subfield is used for healthcare eligibility inquiry transactions, the use of the Additional Amount and Amount Type fields can vary depending on whether the merchant supports optional amount types. If the merchant does not support

optional amount types, a single occurrence of the Additional Amount field represents the total healthcare amount, and the Amount Type field contains the value 4S; no other occurrences of the field are used.

If the merchant supports optional amount types, the message contains up to six occurrences of the Additional Amount field. The first occurrence of the Additional Amount field represents the total healthcare amount, and the Amount Type field is set to 4S. In subsequent occurrences of the subfield, the Amount Type field contains a value specific to each charge (4S, 4U, 4V, 4W, or 4X) that make up the total. For example, if a purchase includes $15 (USD) of over the counter and $30 prescriptions, there would be three occurrences of the subfield, as shown below:

Occurrence	Amount Type Field	Amount Field
1	4S	$45.00
2	4S	$15.00
3	4U	$30

SFID m — Healthcare Service Data

Request: Optional. Variable length of 19 to 95 bytes.

Response: Optional. Variable length of 19 to 95 bytes.

This subfield contains information associated with healthcare eligibility inquiry transactions. The format is as follows:

Service	occurs 1 to 5 times
Provider ID	PIC X(9)
Type Code	PIC X(2)
Payer ID	PIC X(6)
Reason Code	PIC X(2)

These fields are further described below.

Position	Length	Description
01-95	19-95	**Service (occurs 1 to 5 times)**
01-09	9	**Provider ID** The medical license number of the healthcare service provider.
10-11	2	**Type Code** The healthcare service type code as defined by the Health Insurance Portability and Accountability Act (HIPAA).
12-17	6	**Payer ID** The health insurance carrier/payer ID.
18-19	2	**Reason Code** The eligibility approval or rejection reason code as defined by the HIPAA.

SFID n — Error Flag

Request: Not applicable.

Response: Optional. Fixed length of 1 byte

This subfield contains a code providing additional information about the disposition of the transaction. Valid values are as follows:

A = Host adjustment limit exceeded
C = Card verification failed
E = Host return limit exceeded
I = Invalid MAC
K = KMAC synchronization error
L = Invalid PIN length
M = MAC failure
P = Invalid PIN block
R = Sanity check error—previous zone
S = Sanity check error
T = Host system error (token error)
U = Recurring payment cancellation service
V = Stop payment order

W	=	Revocation of authorization order
X	=	Revocation of all authorizations order
1	=	New account information available for recurring payments transaction
2	=	Try again later, recurring payments transaction
3	=	Do not try again for recurring payments transaction
blank	=	No information available

SFID o — American Express Additional Data

Request: Optional. Variable-length of up to 300 bytes.

Response: Not applicable.

This subfield contains additional data for American Express transactions. The subFID is variable-length, and the format is as follows:

Additional Data ID	PIC X(3)
Additional Data	PIC X(297)

These fields are further described below.

Position	Length	Description
01-03	3	**Additional Data ID** Indicates the type of additional data that follows. Valid values are as follows. ITD = Card Not Present Data (mail, telephone and internet order) APD = Airline Passenger Data
04-300	0-297	**Additional Data** Contains either Card Not Present Data or Airline Passenger Data depending on the value of the Additional Data ID field.

Position	Length	Description
04-270	0-267	**Card Not Present Data** The following data fields may be used when the Additional Data ID field is set to a value of ITD. Each data field is preceded by a three-byte data ID and a two-byte data length indicator. Each data ID/data length/data field group is optional, and the groups may occur in any order.
	3	**Data ID** CE$\not b$ = Customer Email ($\not b$ = blank character)
	2	**Data Length** Length of the following field.
	1-60	**Customer Email** Customer's email address.
	3	**Data ID** CH$\not b$ = Customer Host Name ($\not b$ = blank character)
	2	**Data Length** Length of the following field.
	1-60	**Customer Host Name** Name of the server to which the customer is connected.
	3	**Data ID** HBT = HTTP Browser Type
	2	**Data Length** Length of the following field.
	1-60	**HTTP Browser Type** Customer's HTTP browser type.
	3	**Data ID** STC = Ship To Country

Position	Length	Description
	2	**Data Length** Length of the following field.
	3	**Ship To Country** Three-byte numeric ISO country code.
	3	**Data ID** SM$\not b$ = Shipping Method ($\not b$ = blank character)
	2	**Data Length** Length of the following field.
	2	**Shipping Method** Shipping method code. Valid values are as follows: 01 = Same day 02 = Overnight/next day 03 = Priority, 2-3 days 04 = Ground, 4 or more days 05 = Electronic delivery 06–ZZ = Reserved for future use
	3	**Data ID** MPS = Merchant Product SKU
	2	**Data Length** Length of the following field.
	1-15	**Merchant Product SKU** Unique SKU (Stock Keeping Unit) inventory reference number of the product associated with this authorization request. For multiple items, enter the SKU of the most expensive item.
	3	**Data ID** +IP = Customer IP (ACI-defined data ID)
	2	**Data Length** Length of the following field.

Position	Length	Description
	15	**Customer IP** Customer's internet IP address in the following format: nnn.nnn.nnn.nnn
	3	**Data ID** +AN = Customer ANI (ACI-defined data ID)
	2	**Data Length** Length of the following two fields.
	10	**Customer ANI** ANI (Automatic Number Identification). The specified phone number that the customer used to place the order with the merchant.
	2	**Customer II Digits** Telephone company-provided ANI II (Information Identifier) code digits that are associated with the Customer ANI phone number and correspond to the call type (e.g., cellular, government institution).
04-300	0-297	**Airline Passenger Data** The following data fields may be used when the Additional Data ID field is set to a value of APD. Each data field is preceded by a three-byte data ID and a two-byte data length indicator. Each data ID/data length/data field group is optional and the groups may occur in any order.
	3	**Data ID** +DD = Departure Date (ACI-defined data ID)
	2	**Data Length** Length of the following field.
	8	**Departure Date** Departure date in the format YYYYMMDD.

Position	Length	Description
	3	**Data ID** APN = Airline Passenger Name
	2	**Data Length** Length of the following field.
	23-40	**Passenger Name** Passenger name in the following format where ~~b~~ equals a blank character: SURNAME~~b~~ FIRSTNAME~~b~~MIDDLEINITIAL~~b~~TITLE Data must be 23 bytes minimum, left-justified, and blank-filled as needed. 40 bytes maximum, truncate if necessary.
	3	**Data ID** CN~~b~~ = Cardmember Name (~~b~~ = blank character)
	2	**Data Length** Length of the following field.
	23-40	**Cardmember Name** Cardmember name in the following format where ~~b~~ equals a blank character: SURNAME~~b~~FIRSTNAME~~b~~MIDDLEINITIAL~~b~~TITLE Data must be 23 bytes minimum, left-justified and blank-filled as needed. 40 bytes maximum, truncate if necessary.
	3	**Data ID** +AP = Origination/Destination Airport (ACI-defined data ID)
	2	**Data Length** Length of the following two fields.

Position	Length	Description
	5	**Origination Airport** Origination airport for the first segment of the trip. Five-character airport code allows for the anticipated expansion of the current three-character code. Left-justify and blank-fill as needed.
	5	**Destination Airport** Destination airport for the first segment of the trip; not necessarily the final destination. Five-character airport code allows for the anticipated expansion of the current three-character code. Left-justify and blank-fill as needed.
	3	**Data ID** RTG = Routing
	2	**Data Length** Length of the following two fields.
	2	**Number of Cities** Number of airports or cities on the ticket, maximum of 10.
	11-59	**Routing Cities** Routing airport or city code for each leg on the ticket (including Origination Airport and Destination Airport), in five-byte segments each separated by a "/". Example: "ABC␣␣/DEF␣␣/GHI␣␣", where ␣ equals a blank character.
	3	**Data ID** ALC = Airline Carriers
	2	**Data Length** Length of the following two fields.
	2	**Number of Airline Carriers** Number of airline carrier entries in the following field. Maximum of 9.

Position	Length	Description
	5-53	**Airline Carriers** Airline carrier code for each leg on the ticket (including Origination Airport and Destination Airport), in five-byte segments each separated by a "/". Each leg must have an airline carrier code entry, even if multiple (or all) legs are on the same airline. Example: "AB␣␣␣/AB␣␣␣", where ␣ = blank character.
	3	**Data ID** +FR = Fare Data (ACI-defined data ID)
	2	**Data Length** Length of the following three fields.
	24	**Fare Basis** Primary and secondary discount codes indicating the class of service and fare level associated with the ticket. Truncate to 24 bytes, if necessary.
	3	**Number of Passengers** Number of passengers in the party.
	1	**E-Ticket Indicator** Indicates if the ticket is electronic. Valid values are as follows. E = electronic ticket blank = non-electronic ticket
	3	**Data ID** RES = Reservation Code
	2	**Data Length** Length of the following field.
	6-15	**Reservation Code** A precursor to a ticket number. Corresponds to an airline ticket purchase made by an airline or a global distribution system (GDS).

SFID q— EMV Supplementary Request Data

Request: Optional. Variable length of 162 bytes.

Response: Not applicable

The EMV Supplementary Request Data field contains supplementary EMV transaction data. This subfield is optional for all EMV transaction requests. The subfield contains a Dataset ID, followed by a buffer containing a set of EMV data elements, formatted using the Basic Encoding Rules - Tag Length Value (BER-TLV) technique.

For more information about the EMV data elements, refer to the MasterCard M/Chip or the Visa Smart Debit Credit (VSDC) documentation sets or the EMVCo specification.

For Visa contactless transactions, the Dataset ID is 01, and the buffer may contain any of the following EMV data elements:

Name	Tag	Length	Value
Form Factor Indicator	9F6E	04	8 hexadecimal characters
Customer Exclusive Data	9F7C	max 32	Up to 64 hexadecimal characters
Terminal Transaction Qualifiers	9F66	04	8 hexadecimal characters

Other EMV data elements may be included in the subFID, if required.

Note that EMV data elements are typically defined as binary, but the ACI standard POS message supports display-format data only; therefore, each binary byte in an EMV data element must be converted to a pair of hexadecimal characters for transmission in the message. However, there is no change to the value of the "length" subfield, which therefore specifies the number of pairs of hexadecimal characters in the "value" subfield in the ACI standard POS message.

The overall structure is described below:

Position	Length	Description
01–02	2	**Dataset ID** A code indicating the data passed in this subfield. Valid values are as follows: 01 = Visa contactless transaction data
03–162	160	**Buffer** EMV supplementary data, encoded in TLV format.

Note: EMV tags are hexadecimal identifiers for data elements in EMV specifications.

SFID r — Auto-Substantiation Data

Request: Optional. Fixed length of 7 bytes.

Response: Not applicable

This subfield contains information associated with auto-substantiation transactions. The format is as follows:

IIAS Indicator	PIC X
Assigned ID	PIC X(6)
User Field ACI	PIC X(43)

These fields are further described below.

Position	Length	Description
01	1	**IIAS Indicator** An indicator used to specify that an Inventory Informational Approval System (IIAS) was used by the merchant to identify qualified medical items at the point of sale. Valid values are as follows: 0 = An IIAS was not used at the point of sale. 1 = An IIAS was used at the point of sale. 2 = Merchant is exempt from using an IIAS.
02–07	6	**Assigned ID** An identifier assigned to the merchant by MasterCard for IIAS validation. Identifies the merchant for real-time substantiation.
08–50	43	**User Field ACI** Reserved for future use.

Optional Data Subfields — FID 7

The optional data subfields for FID 7 (Product SubFIDs) are summarized in a table below, followed by individual descriptions of the subfields.

Note: The information in FIDs 6 through 9 are contained in subfields. These subfields are included in the request or response message when FIDs 6 through 9 are specified in the host configuration. Specifying FIDs 6 through 9 in the host configuration specifies all subfields for FIDs 6 through 9, respectively. FIDs 6, 7, and 8 are reserved for product use. FID 9 is reserved for customer use.

Summary Table

The subfields for FID 7 are described below according to their subfield identifiers (SFIDs). The table lists the SFID, the picture clause and length of the subfield in the message, and its associated field name. In addition, a check mark (✓) appears in the RQST and RESP columns if a subfield is available for requests and responses, respectively.

SFID	Picture	Length	Field Name	RQST	RESP
a	PIC X(40)	1–40 bytes	Mobile Top-Up Track 2	✓	✓
b	PIC X(15)	15 bytes	Original Mobile Top-Up Reference Number for Refunds (For future use)	✓	
c	PIC X(65)	65 bytes	Mobile Top-Up Response		✓

SFID a — Mobile Top-Up Track 2

Request: Optional. Variable length of 1–40 bytes.

Response: Optional. Variable length of 1–40 bytes.

For requests, this SFID contains the mobile top-up Track 2. This subFID is required for mobile top-up transactions

For responses, this subFID contains the mobile top-up Track 2.

SFID b — Original Mobile Top-Up Reference Number (For future use)

Request: Optional. Fixed length of 15 bytes.

Response: Optional. Fixed length of 15 bytes.

For mobile top-up refund requests, this subFID contains the original top-up reference used to match the refund with the original request.

For mobile top-up refund responses, this subFID contains the original top-up reference used to match the refund with the original request.

SFID c — Mobile Top-Up Response

Request: Not applicable.

Response: Optional. Variable length of 1–65 bytes.

For responses, this subFID contains the pre-pay mobile top-up response. It includes the following fields:

Position	Length	Description
1–16	16	**Top-UP** – Reference number returned for all top-up transactions returned from the mobile operator. or **Activation Code** – For all top-up transactions approved from the inventory stock manager.
17–32	16	**Operator Name** The name of the mobile operator as defined in the Mobile Operator File (MOF).
33–47	15	**Top-Up Approval Code** Present on all approved top-up with purchase and refund top-up with purchase transactions as specified by the mobile operator.

48–65	18	**Other Balance** The available time left on pre-pay phone as specified by the mobile operator or can also be an available balance or a tax amount.

Optional Data Subfields — FID 8

The optional data subfields for FID 8 (Product SubFIDs) are summarized in a table below, followed by individual descriptions of the subfields.

Note: The information in FIDs 6 through 9 are contained in subfields. These subfields are included in the request or response message when FIDs 6 through 9 are specified in the host configuration. Specifying FIDs 6 through 9 in the host configuration specifies all subfields for FIDs 6 through 9, respectively. FIDs 6, 7, and 8 are reserved for product use. FID 9 is reserved for customer use.

Summary Table

The subfields for FID 8 are described below according to their subfield identifiers (SFIDs). The table lists the SFID, the picture clause and length of the subfield in the message, and its associated field name. In addition, a check mark (✓) appears in the RQST and RESP columns if a subfield is available for requests and responses, respectively.

SFID	Picture	Length	Field Name	RQST	RESP
A	PIC X(24)	24 bytes	EBT Voucher Number	✓	
	PIC 9(18)	18 bytes	EBT Available Balance		✓
B	PIC 9(18)	18 bytes	EBT Available Balance		✓

SFID A — EBT Voucher Number or EBT Available Balance

Request: Optional. Fixed length of 24 bytes.

Response: Optional. Fixed length of 18 bytes.

For requests, this subFID can contain the electronic benefits transfer (EBT) voucher number. This subFID is required for manually entered EBT transaction requests that were voice authorized.

For responses, this subFID contains the available balance for a cash account.

Note: Most EBT authorization systems require balances to be printed on a receipt. The terminal may need to format the balance information returned in this subFID for printing on a customer receipt.

SFID B — EBT Available Balance

Request: Not applicable.

Response: Optional. Fixed length of 18 bytes.

In EBT transaction responses, the EBT Available Balance field contains the available balance for a food stamp account.

Note: Most EBT authorization systems require balances to be printed on a receipt. The terminal may need to format the balance information returned in this subFID for printing on a customer receipt.

Request Message Requirements

The customer and vendor have the option of including any number of data fields in requests, as long as the maximum size of the terminal read buffer and the host read buffer are not exceeded. The host read buffer is configurable to a maximum of 4,088 bytes, including all data communications control characters. When the customer determines that any of these data fields must be included in requests for certain transactions, those fields are specified in the host database. The host then enforces these fields as being required. The terminal firmware must also be configured to determine the fields it requires for certain transaction requests.

Standard Message Header

The Standard Message Header is required in every request message sent from a POS device to the host. If a field is not required for a transaction, it still must be accounted for in the header (i.e., it must be blank- or zero-filled). The following table indicates the request message requirements for the standard message header. The first column lists the standard message header fields in the order they must be provided in the request message. The second column of the table lists the transaction request message requirements for each field.

Field	Request Message Requirements
Device Type	Required for all transactions.
Transmission Number	A two-digit number must be provided if the transmission number is checked by the host. If the transmission number is not checked by the host, this field must contain a value of 00 (transmission number not checked).
Terminal ID	Required for all transactions. Must match a value maintained by the host.
Employee ID	Required if the employee ID is validated for the transaction. Otherwise, this field must set to all blank spaces (employee ID not validated).

Field	Request Message Requirements
Current Date	Echoed from response message in terminal-generated reversals. Otherwise, this field is optional. If not used, the field must be zero-filled.
Current Time	Echoed from response message in terminal-generated reversals. Otherwise, this field is optional. If not used, the field must be zero-filled.
Message Type	Set to F or A as indicated by the transaction requested.
Message Subtype	Required.
Transaction Code	Required.
Processing Flag 1	Set to 0 (respond and disconnect) or 1 (respond and do not disconnect).
Processing Flag 2	Set to 5 if sent from an EMV-capable terminal. Set to 1 for passthrough transactions. Otherwise, set to 0.
Processing Flag 3	Set as required for clerk totals transactions. Otherwise, set to 0 (all employee totals for a specific terminal).
Response Code	Set to 000 (response code is not applicable in requests).

Optional Data Fields for Requests

The optional data fields available for requests are listed in the "Optional Data Field Structures" topic documented earlier in this section. Apart from the standard message header, the fields in request messages can be in any order, and subfields must be subordinate to their primary data field. If a field other than those listed in the table described in the "Optional Data Field Structures" topic is submitted in a request and the field identifier is an ASCII alphabetic character, the host disregards the field and continues processing. If the unexpected field identifier is anything other than an ASCII alphabetic character, the request is declined with a response code of 800, indicating a format error occurred.

Transactions Generated with a Credit or Debit Card

Some transaction types require that certain optional fields be used in order for processing to occur. The following list shows the transaction types and the optional fields (from the list described earlier), if any, that are mandatory. This is important information for customers deciding which fields to include in their messages.

The TTC heading in the following table identifies the terminal message type and transaction codes. The message type is either an F or an A, followed by a two-digit transaction code (e.g., F00, F04, A50, A65). An F indicates the transaction is a financial transaction, and an A indicates the transaction is an administrative transaction. The second column of the table provides a text description of each transaction. The third column lists all of the mandatory FIDs and subFIDs for each transaction. No punctuation is placed between FIDs. Mandatory subFIDs are shown in parentheses immediately following the FID to which they belong. For example, Bq represents FID B and FID q and LN26(b) represents FIDs L, N, and 2, and subFID b in FID 6.

Note: Either FID q or FID 2 is required for all financial transactions. The customer selects the preferred track in the host database. The host checks each financial transaction for the presence of either FID q or FID 2 and declines those transactions that have neither FID.

TTC	Description	Mandatory FIDs (SubFIDs)
F00	Normal purchase	Bq or B2
	Purchase from cash account or food stamp account	BDq8(A) or BD28(A)*
F01	Preauthorization purchase	Biq or Bi2
	Preauthorization purchase from cash account	BDiq8(A) or BDi28(A)*
F02	Preauthorization purchase completion	Biq or Bi2
	Preauthorization purchase completion from cash account	BDiq8(A) or BDi28(A)*
F03	Mail or telephone order	Bq

TTC	Description	Mandatory FIDs (SubFIDs)
F04	Merchandise return	Bq or B2
	Merchandise return for a cash account or food stamp account	BDq8(A) or BD28(A)*
F05	Cash advance	Bq or B2
	Cash only (advance) from cash account	BDq8(A) or BD28(A)*
F06	Card verification	q or 2
F07	Balance inquiry	q8(A) or 28(A)*
	Balance inquiry from cash account or food stamp account	Dq8(A) or D28(A)*
F08	Purchase with cash back	BCq or BC2
	Purchase with cash back from cash account	BCDq8(A) or BCD28(A)*
F09	Check verification	LNq or LN2 LNq6(b) or LN26(b)†‡
F10	Check guarantee	BLNq or BLN2 BLNq6(b) or BLN26(b)†‡
F11	Purchase adjustment	BCq or BC2
F12	Merchandise return adjustment	BCq or BC2
F13	Cash advance adjustment	BCq or BC2
F14	Cash back adjustment	BCq or BC2
F15	Card activation	Bq or B2
F16	Additional card activation	Bq6(S) or B26(S)
F17	Replenishment	Bq or B2

TTC	Description	Mandatory FIDs (SubFIDs)
F18	Full redemption	Bq or B2
A50	Logon request	
A51	Logoff request	
A60	Close batch request	
A61	Close shift request	
A62	Close day request	
A64	Clerk totals request	
A65	Batch totals request	
A66	Shift totals request	
A67	Day totals request	
A70	Read mail request	V
A71	Mail delivered request	V
A75	Send mail request	W
A90	Download request	V[§]
A95	Handshake request	

[*] FID 8, subFID A is required only for electronic benefit transfer (EBT) transaction requests and reversals that have been voice authorized.

[†] FID 6, subFID b is required only for electronic check authorization transactions.

[‡] FIDs q or 2 are required for a debit or credit card. Otherwise, FID N (Customer ID) is required.

[§] FID V is required except on initial download requests.

Note: If the terminal can send its own batch, shift, and day totals, the host can also compare the terminal totals to the totals maintained at the host. In this case, FIDs l, o, and m would be sent in close batch, close shift, and close day requests, respectively.

Transactions Generated with an EMV Chip Card

The Processing Flag 2 field in the standard message header must be set to a value of 5 in all transaction requests sent to the host from EMV-capable terminals. For EMV transactions, certain FIDs and subFIDs are required, depending on the type of transaction requested and the type of EMV processing to be performed (i.e., cryptogram authentication only or cryptogram authentication and script processing). Note that some FIDs and subFIDs are always optional. The vendor and customer must coordinate which of these FIDs and subFIDs to use.

The following list shows the transaction types and the optional fields (from the list described earlier), if any, that are mandatory. This is important information for customers deciding which fields to include in their messages.

The TTC heading in the following table identifies the terminal message type and transaction codes. The message type is always an F for EMV transactions, followed by a two-digit transaction code (e.g., F00, F04, F14). An F indicates the transaction is a financial transaction. Administrative transactions are not performed with EMV chip cards. The second column of the table provides a text description for each transaction. The third column lists the mandatory FIDs and subFIDs for each transaction. No punctuation is placed between FIDs. Mandatory subFIDs are shown in parentheses immediately following the FID to which they belong. For example, 6(E) indicates subFID E in FID 6 and 6(EIO) indicates subFIDs E, I, and O in FID 6.

Note: FID 6, subFIDs N and P are optional for all transactions generated with an EMV chip card.

TTC	Description	Mandatory FIDs (SubFIDs)
F00	Normal purchase	Bq6(E) or Bq6(EIO)*
F01	Preauthorization purchase	Biq6(E) or Biq6(EIO)*

TTC	Description	Mandatory FIDs (SubFIDs)
F02	Preauthorization purchase completion	Biq6(E) or Biq6(EIO)*
F03	Mail or telephone order	Bq6(E) or Bq6(EIO)*
F04	Merchandise return	Bq6(E) or Bq6(EIO)*
F05	Cash advance	Bq6(E) or Bq6(EIO)*
F06	Card verification	q6(E) or q6(EIO)*
F07	Balance inquiry	q6(E) or q6(EIO)*
F08	Purchase with cash back	BCq6(E) or BCq6(EIO)*
F11	Purchase adjustment	BCq6(E) or BCq6(EIO)*
F12	Merchandise return adjustment	BCq6(E) or BCq6(EIO)*
F13	Cash advance adjustment	BCq6(E) or BCq6(EIO)*
F14	Cash back adjustment	BCq6(E) or BCq6(EIO)*

* FID 6, subFIDs I and O are required only if cryptogram authentication is to be performed by the host for the transaction.

Request Field Examples

The following examples represent various request configurations. In the examples, a period (.) represents a field separator character (hex 1C), and an exclamation point (!) represents a record separator character (hex 1E). The request data shown below is assumed to follow the standard message header and to be terminated with an ETX in each situation. The quotation marks (“ ”) are used only to delimit the text and are not part of the request. Each example is set up in two parts. The first part identifies and explains the fields included in the request example. The second part illustrates the request format.

Track 2 with Amount 1

The following table describes the fields in the request message when Track 2 data and the transaction amount are included in the request. In the example, Track 2 data was manually entered, so the terminal builds the request as follows:

Message Data	Description
.q	FID associated with Track 2 data
M	Entry ID (M is for manually entered Track 2 data)
1234567890123456789	Primary account number (PAN)—maximum 19 characters
=	Separator character—delimits the PAN
0108	Card expiration date (YYMM)
0	Member number—optional
?	End sentinel
.B	FID associated with the Amount 1 field
12345	The amount of the transaction (e.g., $123.45 or £123.45)

Using the values set in the table above, FIDs q and B for this request would be formatted as follows:

```
Request ".qM1234567890123456789=01080?.B12345"
```

Track 2, Customer PIN, Amount 1, and Sequence Number

The following table is an example where Track 2 data, the customer PIN, the transaction amount, and the transaction sequence number are included in the request. In the example, Track 2 data was manually entered, so the terminal builds the request as follows:

Message Data	Description
.q	FID associated with Track 2 data
M	Entry ID (M is for manually entered Track 2 data)
1234567890123456789	Primary account number (PAN)—maximum 19 characters
=	Separator character—delimits the PAN
0108	Card expiration date (YYMM)
0	Member number—optional
?	End sentinel
.b	FID associated with the customer PIN
8747	Customer PIN
.B	FID associated with the Amount 1 field
1000	The amount of the transaction (e.g., $10.00 or £10.00)
.h	FID associated with the Sequence Number field
0020030120	The sequence number in the following format: shift number—002 batch number—003 sequence number—012 reset flag—0

Using the values set in the table above, FIDs q, b, B, and h for this request would be formatted as follows:

```
Request ".qM1234567890123456789=01080?.b8747.B1000.h0020030120"
```

Track 1, Amount 1, and Host Original Data

The following table is an example where Track 1 data, the transaction amount, and host original data are included in the request. In the example, the terminal builds the request as follows:

Message Data	Description
.2	FID associated with customer Track 1 data
%	Track 1 data start sentinel
B	Format code
1234567890123456789	Primary account number (PAN)
^	Track 1 field separator
JSMITH	Cardholder name
^	Track 1 field separator
0108	Card expiration date (YYMM)
000	Service code
?	Track 1 end sentinel
.B	FID associated with the Amount 1 field
1000	The amount of the transaction (e.g., $10.00 or £10.00)
.6	FID associated with the product subfields
!A	SFID associated with Host Original Data
101512300908	Host original data—time (hhmmsshh) and date (MMDD)

Using the values set in the table above, FIDs 2, B, and 6 (subFID A) for this request would be formatted as follows:

```
Request ".2%B1234567890123456789^JSMITH^0108000?.B1000.6!A101512300908"
```

Track 2, Amount 1, Host Original Data, and Customer Manual Card Verification Digits (CVD)

The following table is an example where Track 2 data, the transaction amount, host original data, and customer manual card verification digits (CVD) are included in the request. In the example, the terminal builds the request as follows:

Message Data	Description
.q	FID associated with Track 2 data
M	Entry ID (M is for manually entered Track 2 data)
1234567890123456789	Primary account number (PAN)
=	Separator character—delimits the PAN
0108	Card expiration date (YYMM)
0	Member number
?	End sentinel
.B	FID associated with the Amount 1 field
1000	The amount of the transaction (e.g., $10.00 or £10.00)
.6	FID associated with product subfields
!A	SFID associated with host original data
101512300908	Host original data—time (hhmmsshh) and date (MMDD)
!B	SFID associated with customer manual CVD
1234	Customer manual CVD

Using the values set in the table above, FIDs q, B, and 6 (subFIDs A and B) for this request would be formatted as follows:

```
Request ".qM1234567890123456789=01080?.B1000.6!A101512300908!B1234"
```

Response Message Requirements

As with requests, the customer and vendor have the option of including any number of data fields in responses, as long as the maximum size of the terminal read buffer and the host read buffer are not exceeded. The host read buffer is configurable to a maximum of 4,088 bytes, including all data communications control characters. When the customer determines that any of these data fields must be included in responses for certain transactions, those fields are specified in the host database. The host then enforces these fields as being required. The terminal firmware must also be configured to determine the fields it requires for certain transaction responses.

Standard Message Header

The Standard Message Header is required in every response message sent from the host to an ACI standard POS device. If a field is not required for a transaction, it still must be accounted for in the header (i.e., it must be blank- or zero-filled). The following table indicates the response message requirements for the standard message header. The first column lists the standard message header fields in the order they must be provided in the response message. The second column of the table lists the transaction response message requirements for each field.

Field	Response Message Requirements
Device Type	Echoed from request.
Transmission Number	Echoed from request.
Terminal ID	Echoed from request.
Employee ID	Echoed from request.
Current Date	The current system date of the host (in YYMMDD format) taking into account different time zones.
Current Time	The current system time of the host (in hhmmss format) taking into account different time zones.
Message Type	Echoed from request.
Message Subtype	Required.

Field	Response Message Requirements
Transaction Code	Echoed from request.
Processing Flag 1	Set to 0 (no mail waiting) or 1 (mail waiting).
Processing Flag 2	Set to 0 (no download waiting) or 1 (download waiting).
Processing Flag 3	Set to 0 (not applicable in response messages).
Response Code	Set according to the authorization processing result.

Optional Data Fields for Responses

The optional data fields available for responses are listed in the "Optional Data Field Structures" topic documented earlier in this section. The order of the fields in the response messages is according to FIDs, with numbered FIDs after lettered FIDs, and uppercase FIDs before lowercase FIDs. SFIDs are always subordinate to their primary FID, and they follow the same hierarchy. Some fields have initialized values. For example, if the card data for the request PAN cannot be located, any related fields such as application account numbers, contain initialized values or zeros.

Some fields may be included in responses regardless of the way the host is configured, including: FID H, the Authentication Key; and FID V, the Mail/Download Key. However, these FIDs are returned only when found to be applicable. The Authentication Key is included when a new MAC communications key is generated due to the configured number of consecutive messages failing to be verified when using message authentication codes (MACs).

If the host is unable to format a response message due to host configuration errors, the response message consists of a header only, with the response code set to 821, indicating an invalid response length. A possible error in the host configuration could be that the format specified results in a response longer than the maximum allowable response.

Note: FIDs V and W are the only FIDs allowed in the response message for a download.

Transactions Generated with a Credit or Debit Card

Some transaction types require that certain optional fields be used in order for processing to occur. The following list shows the transaction types and the optional fields (from the list described earlier), if any, that are mandatory. This is important information for customers deciding which fields to include in their messages.

The TTC heading in the following table identifies the terminal message type and transaction codes. The message type is either an F or an A, followed by a two-digit transaction code (e.g., F00, F04, A50, A65). An F indicates the transaction is a financial transaction, and an A indicates the transaction is an administrative transaction. The second column of the table provides a text description of each transaction. The third column lists all of the mandatory FIDs and subFIDs for each transaction. No punctuation is placed between FIDs. Mandatory subFIDs are shown in parentheses immediately following the FID to which they belong. For example, VW represents FID V and FID W and 6(S) represents FID 6, subFID S.

Note: The terminal vendor and customer should determine whether FIDs and subFIDs in requests are echoed back in responses. The terminal must also be able to process any number of additional FIDs and subFIDs returned in a response. For example, batch totals (FID l), day totals (FID m), and shift terminal totals (FID o) can be returned in any response from the host. Since FIDs l, m, and o can be returned in any transaction response from the host, they are not shown in the optional FIDs column of the table.

TTC	Description	Mandatory FIDs (SubFIDs)
F00	Normal purchase	
	Purchase from cash account or food stamp account	
F01	Preauthorization purchase	
	Preauthorization purchase from cash account	
F02	Preauthorization purchase completion	
	Preauthorization purchase completion from cash account	
F03	Mail or telephone order	

TTC	Description	Mandatory FIDs (SubFIDs)
F04	Merchandise return	
	Merchandise return for a cash account or food stamp account	
F05	Cash advance	
	Cash only (advance) from cash account	
F06	Card verification	
F07	Balance inquiry	
	Balance inquiry from cash account or food stamp account	
F08	Purchase with cash back	
	Purchase with cash back from cash account	
F09	Check verification	
F10	Check guarantee	
F11	Purchase adjustment	
F12	Merchandise return adjustment	
F13	Cash advance adjustment	
F14	Cash back adjustment	
F15	Card activation	6(S)
F16	Additional card activation	6(S)
F17	Replenishment	6(S)
F18	Full redemption	6(S)
A50	Logon response	
A51	Logoff response	
A60	Close batch response	

TTC	Description	Mandatory FIDs (SubFIDs)
A61	Close shift response	
A62	Close day response	
A64	Clerk totals response	n
A65	Batch totals response	l
A66	Shift totals response	o
A67	Day totals response	m
A70	Read mail response	VW
A71	Mail delivered response	
A75	Send mail response	
A90	Download response	VW
A95	Handshake response	

Transactions Generated with an EMV Chip Card

For EMV transactions, certain FIDs and subFIDs are required in response messages, depending on the type of transaction requested and the type of EMV processing to be performed (i.e., cryptogram authentication only, cryptogram authentication and script processing). Note that some FIDs and subFIDs are always optional. The vendor and customer must coordinate which of these FIDs and subFIDs to use.

The following list shows the transaction types and the optional fields (from the list described earlier), if any, that are mandatory. This is important information for vendors and customers deciding which fields to include in their messages.

The TTC heading in the following table identifies the terminal message type and transaction codes. The message type is always an F for EMV transactions, followed by a two-digit transaction code (e.g., F00, F04, F14). An F indicates the transaction is a financial transaction. Administrative transactions are not used with EMV chip cards. The second column of the table provides a text description for each transaction. The third column lists the mandatory FIDs and subFIDs for each

transaction response. No punctuation is placed between FIDs. SubFIDs are shown in parentheses immediately following the FID to which they belong. For example, 6(QR) indicates FID 6, subFIDs Q and R.

Note: FID 6, subFIDs E and I are optional for all EMV transaction response messages.

TTC	Description	Mandatory FIDs (SubFIDs)
F00	Normal purchase	6(QR)*
F01	Preauthorization purchase	6(QR)*
F02	Preauthorization purchase completion	6(QR)*
F03	Mail or telephone order	6(QR)*
F04	Merchandise return	6(QR)*
F05	Cash advance	6(QR)*
F06	Card verification	6(QR)*
F07	Balance inquiry	6(QR)*
F08	Purchase with cash back	6(QR)*
F11	Purchase adjustment	6(QR)*
F12	Merchandise return adjustment	6(QR)*
F13	Cash advance adjustment	6(QR)*
F14	Cash back adjustment	6(QR)*

* For cryptogram authentication, only FID 6, subFID Q is required. For script processing, both subFIDs Q and R for FID 6 are required.

Response Field Examples

The following examples represent various response configurations. In the examples, a period (.) represents a field separator character (hex 1C), and an exclamation point (!) represents a record separator character (hex 1E). The response data shown below is assumed to follow the standard message header and

to be terminated with an ETX in each situation. The quotation marks (“ ”) are used only to delimit the text and are not part of the response. Each example is set up in two parts. The first part identifies and explains the fields included in the response example. The second part illustrates the response format.

Response Display and Transaction Description

The following table is an example where response display data and a description of the transaction are included in the response. In the example, the response is formatted as follows:

Message Data	Description
.g	FID associated with the Response Display field.
APPROVAL	The text associated with the response display set up in the ACI Standard Device Response File (ARSP)—maximum 48 characters.
.s	FID associated with the Transaction Description field.
PURCHASE FROM CREDIT CRD	A description of the transaction for receipt purposes—maximum 24 characters.

Using the values set in the table above, FIDs g and s for this response would be formatted as follows:

```
Response ".gAPPROVAL.sPURCHASE FROM CREDIT CRD"
```

Batch Totals

The following is an example where batch totals are included in the response. In the example, the response is formatted as follows:

Message Data	Description
.l	FID associated with the Totals/Batch field.
003	Shift number portion of the batch totals.
002	Batch number portion of the batch totals.

Message Data	Description
0023	Number of debits portion of the batch totals.
+123456789012345678	Amount of debits portion of the batch totals. All amount fields are preceded by a sign (+ or -).
0003	Number of credits portion of the batch totals.
+123456789012345678	Amount of credits portion of the batch totals. All amount fields are preceded by a sign (+ or -).
0001	Number of adjustments portion of the batch totals.
+123456789012345678	Amount of adjustments portion of the batch totals. All amount fields are preceded by a sign (+ or -).

Using the values set in the table above, FID l for this response would be formatted as follows:

```
Response ".10030020023+123456789012345678 0003+123456789012345678 0001
+123456789012345678"
```

3: Download Data

The ACI standard POS message supports full and partial downloads to POS terminals. Downloads provide a way to transport and load the data necessary to configure a terminal. Configuration data downloaded to terminals is set up in the host database. This section provides the following information on host download support:

- Download record format
- Download field identifiers (DIDs)
- Data element records
- Full download requests and responses
- Partial download requests and responses

Downloading Data to Terminals

Using the ACI standard POS message, a POS device can request a download at any time. The host informs a POS device when a download is pending in a transaction response, but the device must request download information from the host before it can receive any of the configuration data required for processing. When an ACI standard POS terminal requests download information, the host retrieves the download data from a host database.

The configuration data downloaded to the terminal from the host depends on the request sent from the terminal. Terminals can request a full download, which results in all of the configuration data being sent to the terminal. Terminals can also request a partial download, asking for a single download field.

The data elements to be included within the host database are determined by the customer. Information that can be contained within these database records includes telephone numbers, floor limits, and receipt data. Typically, download database records are organized by terminal group. Thus, the same information can be downloaded to all terminals belonging to the same group.

Download Record Format

When a terminal requests a download, the response from the host must include the Mail/Download Key data element (FID V) and the Mail/Download Text data element (FID W). FIDs V and W provide the processing mechanisms required in order for the host to perform downloads. The functions of these FIDs include initiating the download, notifying the terminal that download data exists, and transporting the download text to the terminal. The following paragraphs describe FID V and FID W in detail.

Download Key (FID V)

The Download Key data element (FID V) is the first block of information formatted in the response to a download request. The download key is used to determine whether a full or partial download was requested.

If a full download is requested, FID V indicates whether this is an initial request or a continuation request. Continuation requests occur when the download is too large to fit into a single message and requires several request and response message exchanges in order to complete. If a partial download is requested, FID V identifies the download field requested.

FID V consists of the following fields:

Category Code — Indicates an initial download or whether the terminal requests additional download data.

The Category Code consists of two bytes. The first byte is used in responses and the second byte is used in requests.

In responses, the Category Code indicates whether all of a particular download field has been loaded or if more download data exists. If the value in the first byte of the Category Code is 1, this indicates to the terminal that more data exists for a particular download field. If the first byte of the Category Code is 0, this indicates to the terminal that no more data exists for the particular download field. In this case, the next field is downloaded.

In requests, the Category Code indicates whether the terminal requests to receive remaining download data (as indicated in the response) or begin receiving information from the next field to be downloaded. If the second byte of the Category Code is 1, this indicates that the terminal is requesting to receive the remaining data for the download field identified as incomplete in the response (indicated by 1 in the first byte of the Category Code). The terminal can continue to receive the balance of the data for the incomplete download field until no more data exists as long as the second byte of the Category Code is 1. When no more data exists, the next field is downloaded. If the second byte of the Category Code is 0, this indicates that the terminal is requesting to begin receiving information from the next field to be downloaded.

Access Code — A one-byte flag indicating whether this is the initial download request, a download continuation request, or a partial download request. Valid values for downloads are as follows:

1 = Initial download request
2 = Download continuation request
5 = Read specific download field

Processing Flag — A two-byte flag indicating the DID of the current or next download field included in the response. For full downloads, the DID is echoed. For partial downloads, this flag identifies the DID to be downloaded.

Filler — Ten bytes filled with zeroes. The value in this field is not used for downloading.

FID V precedes FID W in download responses from the host to the terminal. A field separator (FS) of a period (.) is used to identify where FID V begins and ends. FID V is formatted in the following manner:

FS	FID V	Category Code	Access Code	Processing Flag	Filler	FS

Note: The fields associated with FID V are further explained later in this section in the topical discussions of "Full Downloads" and "Partial Downloads."

Download Text (FID W)

The Download Text data element (FID W) contains the download field identifier (DID) and the download text associated with the DID. This is the configuration data sent to the terminal. FID W consists of the following fields:

Destination DPC Number — This field is not used in downloads, but contains a value of 0.

Download Data Text — The device configuration data being downloaded to the terminal. This includes the DID and the associated text. Up to 448 bytes can be included in this field.

FID W follows FID V in download responses from the host to the terminal. Configuration data is taken from the host database and formatted as shown below. A total of 82 data elements can be included within the format, although only 68 are currently supported (14 DIDs are reserved for future use). A field separator (FS) of a period (.) is used to identify where FID W begins and ends. Within FID W, every data element included is prefixed with a group separator character of a comma (,), a download field identifier (DID) of one character, and a space. All these data elements are included within the optional data field of FID W.

FS	FID W	Destination DPC Number	Download Text	FS

Defining Data Elements

Data elements are defined and set up by customers in the host database. Each data element that can be downloaded to the terminal is identified with a download field identifier (DID). The range of identified DIDs is A through Z, 0 through 29, and a through z.

The data elements identified by DIDs are sent only if they are requested by the terminal and if they contain data. If the terminal requests these data elements, but the data elements contain no data, they are not sent to the terminal.

The table shown below identifies each DID and provides a brief description of the type of information that can be entered for each DID.

DIDs	Description
A–Z	DIDs A–Z can consist of any information the terminal owner and operator want to include or the terminal vendor requires. These DIDs are optional. Each of these DIDs can contain up to 40 alphanumeric characters. An example of data that might be included in these data elements is receipt information. Customers can configure receipts differently for various terminals or terminal groups.
0–29	DIDs 0–29 contain card prefix ranges and associated card prefix range information. The format for each of these DIDs is provided later in this section under the topic “Card Prefix Information.”
a–z	DIDs a–z (lower case a through z) contain specific processing controls from the host database. The format for each of these DIDs is provided later in this section under the topic “Processing Controls.”

Card Prefix Information

DIDs 0–29 consist of card prefix range information defined by the terminal owner and operator. Up to 30 card prefix ranges can be defined in these DIDs, with one card prefix range defined in each DID.

Data Element Structures

The following table describes the data element structure of the card prefix information included in DIDs 0–29. The table includes the positions in the DID occupied by each card prefix data element, a picture clause containing the field length and format of the card prefix data element, and the name of the card prefix data element. The card prefix information consists of 108 alphanumeric characters per prefix, organized in the following format. Each of the data elements contained in this table is described in detail following the table.

Card Prefix Data Element Structures		
Position	**Picture**	**Data Element Name**
1–11	PIC X(11)	Low Prefix
12–22	PIC X(11)	High Prefix
23–42	PIC X(20)	Main Network Telephone Number
43–62	PIC X(20)	Backup Network Telephone Number
63–82	PIC X(20)	Referral Telephone Number
83–94	PIC X(12)	Retailer ID
95	PIC X(01)	Draft Capture Flag
96	PIC X(01)	Totals Flag
97	PIC X(01)	PIN Validation Flag
98	PIC X(01)	Receipt Flag
99	PIC X(01)	MOD-10 Check Flag
100	PIC X(01)	PAN Fraud Check Flag
101–108	PIC X(08)	User Defined Data

Data Element Descriptions

The following paragraphs provide descriptions of the data elements that can be downloaded in DIDs 0–29. The descriptions include any values associated with the data element.

Low Prefix — The lowest card prefix value defined by this range. Any card prefix values greater than or equal to the value specified in this field, and less than or equal to the value specified in the High Prefix field are included within the card prefix range defined.

High Prefix — The highest card prefix range defined by this range. Any card prefix range values less than or equal to the value specified in this field, and greater than or equal to the value specified in the Low Prefix field are included within the card prefix range defined.

Main Network Telephone Number — The telephone number of the primary network associated with transactions initiated with cards in the defined card prefix range.

Backup Network Telephone Number — The telephone number of the backup network associated with transactions initiated with cards in the defined card prefix range.

Referral Telephone Number — The telephone number of the network designated to receive referrals concerning transactions initiated with cards in the defined card prefix range.

Retailer ID — An identifier for the retailer associated with this card prefix range. The retailer ID is assigned by MasterCard, Visa, American Express, or another card issuer.

Draft Capture Flag — A flag indicating whether cards within the defined range use draft capture processing. Valid values are as follows:

0 = Authorize only
1 = Authorize and capture

Totals Flag — A flag indicating whether the terminal maintains totals for the card prefix range. The terminal must have the capability of maintaining totals if the value entered in this field is 1. Valid values are as follows:

0 = No, do not maintain totals for this card prefix range.
1 = Yes, maintain totals for this card prefix range.

PIN Validation Flag — A flag indicating whether cards within the card prefix range require PINs. Valid values are as follows:

0 = No, PINs are not required for this card prefix range.
1 = Yes, PINs are required for this card prefix range.

Receipt Flag — A flag indicating whether transactions initiated with cards within the card prefix range require a receipt. Valid values are as follows:

0 = No, receipts are not required for this card prefix range.
1 = Yes, receipts are required for this card prefix range.

MOD-10 Check Flag — A flag indicating whether MOD-10 checks are used for cards within the card prefix range. Valid values are as follows:

0 = No, terminal MOD-10 checks are not required for this card prefix range.
1 = Yes, terminal MOD-10 checks are required for this card prefix range.

PAN Fraud Check Flag — A flag indicating whether the terminal performs PAN fraud checks on cards within the card prefix range. If PAN fraud checks are performed, this field also indicates the type. Valid values are as follows:

0 = No PAN fraud check is required for this card prefix range
1 = Visually check the last four digits of the PAN
2 = Visually check Track 2

User Defined Data — Up to eight characters of user-defined data can be entered in this field. This data is used by the terminal for cards within the card prefix range.

Processing Controls

DIDs a–z contain terminal processing control information retrieved from the host database. These DIDs include such information as the PIN encryption key and message authentication code (MAC) key, terminal location, allowed transactions, card processing parameters, and adjustment and return limits.

Currently, 13 of DIDs a through z are used. The DIDs not currently in use are reserved for future use.

Data Element Structures

The following table describes the structure of the data elements associated with DIDs a through z. The table includes the DID associated with each data element, a picture clause containing the field length and format of the data element, and the name of the data element. The data elements contained in DIDs a–z are organized in the following format. Each of the data elements contained in this table is described in detail following the table.

Processing Controls Data Element Structures		
DID	**Picture**	**Data Element Name**
a	PIC X(25)	Terminal Location
b	PIC X(16)	Terminal City and State
c	PIC X(22)	Terminal Owner
d	N/A	Reserved
e	N/A	Reserved
f	PIC X(88)	Service Representative Information
g	PIC X(48)	PIN Encryption Key
h	PIC X(48)	MAC Key
i	PIC X(01)	PIN Pad Character
j	PIC X(48)	Data Encryption Key
k	PIC X(1170)	Service (occurs 30 times)

Processing Controls Data Element Structures		
DID	**Picture**	**Data Element Name**
l	PIC X(44)	Limits
m	N/A	Reserved
n	N/A	Reserved
o	N/A	Reserved
p	PIC X(30)	Allowed Transactions
q	PIC X(19)	Retailer ID
r	PIC X(40)	Merchant Name
s	PIC X(20)	Referral Telephone Number
t	N/A	Reserved
u	N/A	Reserved
v	N/A	Reserved
w	N/A	Reserved
x	N/A	Reserved
y	N/A	Reserved
z	N/A	Reserved

Data Element Descriptions

The following paragraphs provide descriptions of the data elements that can be downloaded in DIDs a–z. The descriptions include any values associated with the data element. For elements that consist of multiple fields, the structure of the elements are shown below. Included for each field is the field position in the element, the field length, and a description of its contents if necessary.

Terminal Location — DID a contains the terminal name/description for printing on receipts in compliance with Regulation E.

Terminal City and State — DID b contains the city and state in which this terminal is located for printing receipts in compliance with Regulation E. The following fields are associated with this data element.

Position	Length	Description
1–13	**13**	City
14–16	**3**	State

Terminal Owner — DID c contains the name of the retailer who owns the terminal.

Reserved — DID d is reserved for future use.

Reserved — DID e is reserved for future use.

Service Representative Information — DID f contains the name, address, and telephone number of the service representative to contact when problems occur with this terminal. The following fields are associated with this data element.

Position	Length	Description
01–25	**25**	**Name**
26–50	**25**	**Address**
51–63	**13**	**City**
64–66	**3**	**State**
67–68	**2**	**Country**
69–88	**20**	**Phone**

PIN Encryption Key — DID g contains the PIN encryption key used by this terminal. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

MAC Key — DID h contains the message authentication code (MAC) generation key used by this terminal. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

PIN Pad Character — DID i contains the character used to pad the PIN block.

Data Encryption Key — DID j contains the data encryption key used by this terminal. The number and type of keys used determines the length of this field, as follows.

- A single-length key contains 16 bytes.
- A double-length key contains 32 bytes.
- A triple-length key contains 48 bytes.

Service (occurs 30 times) — DID k contains the card processing parameters for this terminal.

If the data in DID k exceeds the maximum response length supported by the host application, the host processes the downloading of this DID as follows. When DID k is configured to be included in a full download to a terminal, the host adds as many occurrences as will fit to the current response. The host then sends the response to the terminal with the first byte of FID V (Main/Download Key) set to a value of 1 to indicate to the terminal that more data remains in this field. Next, the terminal can request the remaining portion of this field by changing the second byte of FID V to a value of 1 and using the resulting FID V in its next request, or the terminal can proceed to the next DID by echoing FID V unaltered. The following structure represents one occurrence of this field. This field can occur up to 30 times.

Position	Length	Description
01–02	**2**	**Card Type** Indicates the type of card accepted at this terminal.
03–11	**9**	**Normal Purchase Floor Limit** The floor limit used by the host, in whole currency units, for normal purchase transactions performed at this terminal for this card type.
12–20	**9**	**Cash Advance Floor Limit** The floor limit used by the host, in whole currency units, for cash advance transactions performed at this terminal for this card type.
21–29	**9**	**Mail or Telephone Order Floor Limit** The floor limit used by the host, in whole currency units, for mail/phone order transactions performed at this terminal for this card type.
30–38	**9**	**Transaction Limit** The transaction amount limit for this card type used by the host, in whole currency units, for this terminal. Transactions for amounts exceeding this limit are denied. This limit does not apply to cards with VIP status.

Position	Length	Description
39	1	**Transaction Profile** A value indicating the transaction profile. Valid values are as follows: 0 = Authorize only with paper follow-up 1 = Authorize and draft capture 2 = Authorize and expect electronic follow-up 3 = Terminal determines data capture mode for each transaction

Limits — DID l contains the limit parameters for the terminal.

Position	Length	Description
01–04	4	**Adjustment Count Limit** The maximum number of adjustment transactions allowed to be performed on this terminal for each terminal cutover-to-cutover period.
05–22	18	**Adjustment Amount Limit** The maximum amount, in whole and fractional currency units, that can be accepted at this terminal using adjustment transactions. This limit is invoked for each terminal cutover-to-cutover period.
23–26	4	**Return Count Limit** The maximum number of merchandise return transactions allowed to be performed on this terminal during each terminal cutover-to-cutover period.
27–44	18	**Return Amount Limit** The maximum amount, in whole and fractional currency units, that can be accepted at this terminal using merchandise return transactions during each terminal cutover-to-cutover period.

Reserved — DID m is reserved for future use.

Reserved — DID n is reserved for future use.

Reserved — DID o is reserved for future use.

Allowed Transactions — DID p contains the following fields used to determine if a transaction is allowed at this terminal. Whenever a transaction is allowed, a 1 is placed in the appropriate field shown below. A 0 indicates the transaction is not allowed. The following list contains the name of the field controlling the transaction. The length of each field shown below is one byte.

Normal Purchase
Preauthorization Purchase
Preauthorization Purchase Completion
Mail or Telephone Order
Merchandise Return
Cash Advance
Card Verification
Balance Inquiry
Purchase With Cash Back
Check Verification
Check Guarantee
Purchase Adjustment
Merchandise Return Adjustment
Cash Advance Adjustment
Close Batch
Close Shift
Close Day
Reserved
Read Mail
Send Mail
Mail Delivered
Sales Drafts
Clerk Totals
Cash Back Adjustment
Adjustments when Amt2 > Amt1
Preauthorizations for a lesser amount
Card Activation
Additional Card Activation
Replenishment
Full Redemption

Retailer ID — DID q contains the identifier of the retailer who owns this terminal as defined in the host database.

Merchant Name — DID r contains the merchant name, as defined in the host database.

Referral Telephone Number — DID s contains the retailer referral telephone number, as defined in the host database.

Reserved — DID t is reserved for future use.

Reserved — DID u is reserved for future use.

Reserved — DID v is reserved for future use.

Reserved — DID w is reserved for future use.

Reserved — DID x is reserved for future use.

Reserved — DID y is reserved for future use.

Reserved — DID z is reserved for future use.

Requesting a Download

When download data is required, the host sets the Processing Flag 2 field in the standard message header to a value of 1 (download waiting). This indicates to the terminal that a download should be requested. The terminal should then programmatically request a full download.

The terminal can also make a request to the host for a partial download. A partial download consists of sending one item to the terminal. This capability is intended for use whenever a single field, such as a telephone number, has changed. In this case, the affected terminals can be notified by mail that they should make a partial download request for the modified field.

Full Downloads

A full download allows a terminal to receive all of the configuration data it requires at once. The terminal can request a full download in order to receive all the configuration data specified in the host database. Responses to a full download request carry as much of the download data that will fit in the response. When a response to a download request does not contain all the configuration data possible, response code 881 is included in the response message, indicating more download data exists. When the terminal receives a response with this response code, the terminal must send a continuation download request to the host in order to receive more configuration data.

A continuation download request uses the same transaction code as the initial download request, but some of the fields associated with FID V, the Download Key, are modified. FID V data consists of the following:

Category Code — Indicates an initial download or whether the terminal requests additional download data.

The Category Code consists of two bytes. The first byte is used in responses and the second byte is used in requests.

In responses, the Category Code indicates whether the last field in FID W contains all the information from that field.

- If the first byte of the Category Code is set to a value of 1, this indicates the last field downloaded in FID W is incomplete. This means the download field could not be fit into a single message given the constraints imposed by the maximum response length configured in the host database. In this case, the download field must be sent in multiple messages. A value of 1 in the first byte of the Category Code indicates to the terminal that more data exists for a particular download field.
- If the first byte of the Category Code is set to a value of 0, this indicates to the terminal that no more data exists for the particular download field. In this case, the next field is downloaded.

In requests, the Category Code indicates whether the terminal is required to receive the remaining information from the last field downloaded in FID W (if applicable) or begin receiving information from the next field to be downloaded.

- The first byte of the Category Code can indicate that more data exists for this download field (DID) by containing a value of 1. If the first byte of the Category Code is set to a value of 1, and the second byte of the Category

Code is also set to a value of 1, this indicates that the terminal is requesting to receive the remaining data for the DID. The terminal can continue to receive the balance of the data for the incomplete download field until no more data exists as long as the second byte of the Category Code is set to a value of 1. When no more data exists, the next field is downloaded.

- If the second byte of the Category Code is set to a value of 0, this indicates that the terminal is requesting to begin receiving information from the next field to be downloaded. This can occur in the following situations:
  - If the terminal is not requesting a continuation of a current download field identified in the response as being incomplete
  - If an incomplete download field is now identified in the response as complete
  - If there was no incomplete download field identified in the response (that is, the download field fit into a single message)

When the download is complete, the Category Code is reset to a value of 00 by the host. For additional information and examples of how the Category Code works, refer to appendix A.

Access Code — A flag indicating whether this is the initial download request, a download continuation request, or a partial download request. Note that an initial download can also be indicated by the absence of FID V. Valid values for downloads are as follows:

1 = Initial download request
2 = Download continuation request
5 = Read specific download field

Processing Flag — The DID of last download field included in the previous response or the DID of the field to be downloaded in a partial download request.

Filler — Ten bytes filled with zeroes. The value in this field is not used for downloading.

Terminal Requests a Full Download

When the terminal requests a full download, FID V should contain the information shown in the table below. FID V consists of four fields, totaling 15 bytes. A value of 1 in the third byte of FID V, the Access Code field, indicates that the terminal is requesting a full download for the first time. DIDs A–Z are downloaded first, followed by the card prefix ranges (DIDs 0–29), and the processing control parameters (DIDs a–z).

FID V–Download Key			
Field	**Format**	**Value**	**Meaning/Origin**
Category Code	PIC X(02)	N/A	N/A
Access Code	PIC X(01)	1	Read first download record
Processing Flag	PIC X(02)	N/A	N/A
Filler	PIC X(10)	0000000000	N/A

The example below illustrates the message format of FID V for a full download request from the terminal.

```
.V001000000000000
```

The key describing the information contained in the example is as follows:

```
.           = Field separator
V           = FID V
00          = Category code (download for a DID completed)
1           = Access code (1 indicates an initial download request)
00          = Processing flag
0000000000  = Filler
```

Host Responds to a Full Download Request

When the terminal sends a download request to the host, the host responds by sending a specific set of information in FID V back to the terminal. FID V consists of four fields, totaling 15 bytes.

If the last field in the message response cannot fit into a single message based on the constraints imposed by the maximum response length configured at the host, the host changes the first byte of the Category Code field to a value of 1. This value indicates to the terminal that the field must be sent in multiple messages. If all of the information in the last field of the message response has been downloaded to the terminal, the host sets the first byte of the Category Code field to a value of 0. The second byte of the Category Code field is used in requests.

A value of 2 in the Access Code field indicates that this is a response to a download request.

The Processing Flag field contains the last DID included in the response. Following FID V, the response contains FID W, with up to 956 characters of download text from the host database.

The following tables show the information contained in FID V and FID W for a response from the host to the terminal.

FID V–Download Key			
Field	**Format**	**Value**	**Meaning/Origin**
Category Code	X(02)	00, 10	00 = Download for a DID completed. 10 = More information exists for this DID.
Access Code	X(01)	2	Continue full download (not initial request)
Processing Flag	X(02)	*did*	Last DID in response; DIDs must be left-justified and blank-filled (e.g., “4 ”)
Filler	X(10)	0000000000	N/A

FID W–Download Text			
Field	**Format**	**Value**	**Meaning/Origin**
DPC Number	N/A	N/A	Not used
Download Text	X(956)	N/A	Maximum = 956 characters

The example below illustrates the message format of FIDs V and W for a full download response from the host to the terminal, where ␣ denotes a blank space. Card prefixes are downloaded after DID Z.

`.V0024␣0000000000.W0,4␣data`

The key describing the information contained in the example is as follows:

.	=	Field separator
V	=	FID V
00	=	Category code (download for a DID completed)
2	=	Access code (2 indicates a download continuation request)
4␣	=	Processing flag (␣ indicates a blank space)
0000000000	=	Filler
.	=	Field separator
W	=	FID W
0	=	Destination DPC
,	=	Group separator
4␣	=	DID 4 (␣ indicates a blank space)
data	=	Download data for card prefix range 4

Continuation of a Full Download Request

Continuation requests from the terminal are sent when the response received from the host indicates that more download information needs to be retrieved. Continuations mean that the last field in the message is incomplete, and that the field cannot fit into a single message based on the constraints imposed by the maximum response length configured at the host. Therefore, the field or remaining fields must be sent in multiple messages.

On receipt of a response with a response code indicating that more download information needs to be retrieved, the terminal sends a specific set of information in FID V. FID V consists of four fields, totaling 15 bytes.

If the last field in the message response cannot fit into a single message based on the constraints imposed by the maximum response length configured at the host, the host changes the Category Code field. The first byte of the Category Code is used in responses from the host to the terminal. The second byte of the Category Code field is used in requests from the terminal to the host. The following processing transpires during a continuation request from the terminal:

1. The host receives a request from the terminal to send download data.
2. The host changes the first byte of the Category Code field to a value of 1 in the response to the terminal. This indicates to the terminal that the field must be sent in multiple messages.
3. If the terminal wants the remaining information to be sent, it changes the second byte of the Category Code field to 1. If the terminal does not require the remaining information for the last field in the response, the second byte of the Category Code field is 0. The terminal formats the request and sends it to the host.
4. The host receives the request. If the terminal wants the remaining download information, the host checks to determine if the remaining information can fit into the response. If it can, the host changes the value in the first byte of the Category Code to 0, indicating no more download data exists. If it cannot, the host leaves the value in the first byte of the Category Code at 1, indicating to the terminal that more download data exists.

   This request and response sequence continues until all of the download information requested by the terminal has been sent.

   Once all of the information has been downloaded for the DID, the host changes the Category Code to 00, indicating the download for the DID is complete. In cases where the message response can fit into a single message, FID V is echoed in the request from the terminal.

A value of 2 in the Access Code field indicates that this is not an initial download request, but rather a continuation request. The last DID included in the response is returned to the host. This information is echoed from the response from the host.

For more information on and an example of how the Category Code works during a continuation request, refer to appendix A.

The following table shows the information contained in FID V for a continuation request from the terminal.

FID V–Download Key			
Field	**Format**	**Value**	**Meaning/Origin**
Category Code	PIC X(02)	00, 10, 11	00 = Download for a DID completed. 10 = No more data exists, or more information exists but the terminal does not request it. 11 = The terminal requests the remaining information.
Access Code	PIC X(01)	2	Read next
Processing Flag	PIC X(02)	*did*	Last DID downloaded; DID must be left-justified and blank-filled
Filler	PIC X(10)	0000000000	N/A

The example below illustrates the message format for a full download continuation request from the terminal, where ␣ denotes a blank space. The last field downloaded to the terminal in its entirety, as indicated by the Category Code, was DID Z.

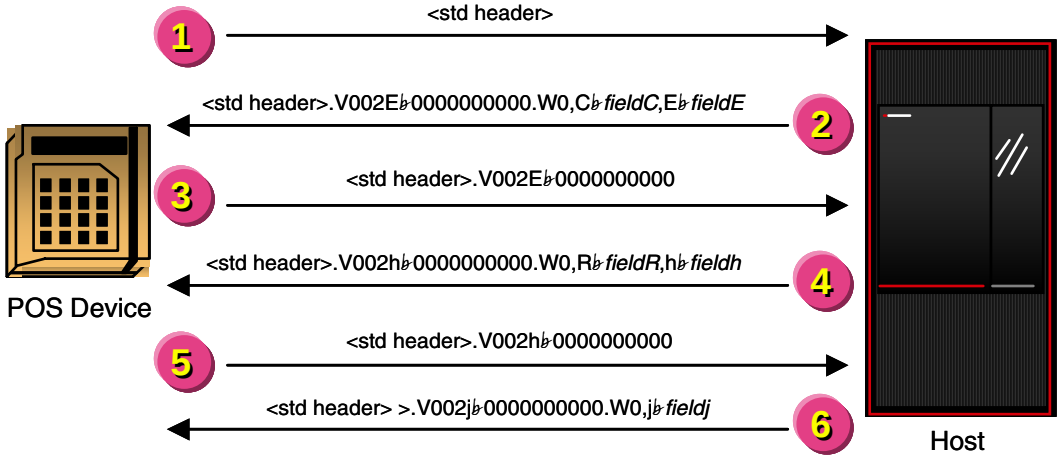
```
.V002Z␣0000000000
```

The key describing the information contained in the example is as follows:

.	=	Field separator
V	=	FID V
00	=	Category code (download for a DID completed)
2	=	Access code (a value of 2 indicates a download continuation request)
Z␣	=	Processing flag (indicates that DID Z was the last data element downloaded)
0000000000	=	Filler

Full Download Example

The example shown below is of a full download. The data shown is assumed to contain a standard header and be terminated with an ETX. The period (.) represents a field separator character that is a hexadecimal representation of 1C. The comma (,) represents a group separator character that is a hexadecimal representation of 1D. The ♭ character represents a blank space. Explanations of the messages are included below.



1. The terminal makes a download request with no optional fields.
2. The host, finding no download key, assumes this to be a full download request. The host formats (into the download text field) as much download data as fits in one response message. In the example, not all data fits in one response. The response code sent in the standard message header is 881, indicating that more data exists.
3. The terminal processes the response. The response code of 881 in the standard message header indicates to the terminal that more download data exists. Therefore, the terminal makes another download request, this time including the download key as received in the prior response.
4. The host uses the DID in the Processing Code field to reinstate itself into the download process to the point it left off, and formats another response with download text and key.
5. The terminal processes the response. The response code of 881 in the standard message header indicates to the terminal that more download data exists. Therefore, the terminal makes another download request, this time including the download key received in the prior response.

6. The above cycle continues until the remaining download data fits within the response message to the terminal. When the last data has been included, the host sets the response code to 880, indicating no more data exists. Note that only those download fields containing values are downloaded. Download fields that do not contain information are not sent to the terminal. In this example, the fields downloaded are C, E, R, h, etc.

Partial Download

The terminal can also make a request to the host for a partial download. A partial download consists of sending one item to the terminal. This capability is intended for use whenever a single field, such as a telephone number, has changed. In this case, the affected terminals can be notified by mail that they should make a partial download request for the modified field.

The difference between a full and a partial download request is that the partial download request must contain FID V to indicate the following:

Category Code — A partial download can be continued in the same manner as a full download if the field exceeds the maximum response length. For more information on how the Category Code is used with partial downloads and an example, refer to appendix A.

Access Code — A value of 5 is placed in the Access Code field to indicate that a particular data field is requested to be downloaded. This value indicates a partial download.

Processing Flag — The DID of the download field requested. The DID must be left-justified and blank-filled.

Filler — Ten bytes filled with zeroes. The value in this field is not used for downloading.

The response to a valid partial download request looks just like the response to a full download request. A response code of 880 indicates that all of the data for the requested DID is contained in the response. A response code of 881 indicates that more data exists for the requested DID and the terminal should request continuation of the download.

Terminal Requests a Partial Download

The terminal formats a download request that includes FID V, the Download Key field, indicating which download field is desired. The following table shows the information contained in FID V for a partial download request from the terminal.

FID V–Download Key (Request Key)			
Field	**Format**	**Value**	**Meaning/Origin**
Category Code	PIC X(02)	00 or 11	00 = Download for DID completed. 11 = Terminal requests remaining information for a DID.
Access Code	PIC X(01)	5	Read specific DID
Processing Flag	PIC X(02)	*did*	DID to be downloaded; DID must be left-justified and blank-filled
Filler	PIC X(10)	0000000000	N/A

The example below illustrates the message format of FID V for a partial download request from the terminal for DID r.

```
.V005rb0000000000
```

The key describing the information contained in the example is as follows:

. = Field separator
V = FID V
00 = Category code (download for a DID completed)
5 = Access code (a value of 5 indicates to read a specific download field)
rb = Processing flag (DID r)
0000000000 = Filler

Host Responds to a Partial Download Request

The host uses FID V, the Download Key, to determine that the request is for a partial download. The host accesses the requested field and responds to the terminal with that field and a response code of 880, indicating that no more data exists. The following tables show the information contained in FID V and FID W for a response from the host to the terminal.

FID V–Download Key (Response Key)			
Field	**Format**	**Value**	**Meaning/Origin**
Category Code	PIC X(02)	00, 10, 11	00 = Download for a DID completed. 10 = No more data exists, or more information exists but the terminal does not request it. 11 = The terminal requests the remaining information for a DID.
Access Code	PIC X(01)	5	Read next (not initial request)
Processing Flag	PIC X(02)	*did*	Last DID in response; DID must be left-justified and blank-filled
Filler	PIC X(10)	0000000000	N/A

FID W–Download Text			
Field	**Format**	**Value**	**Meaning/Origin**
DPC Number	0	N/A	N/A
Text	PIC X(956)	N/A	Maximum is 956 characters

The example below illustrates the message format for a partial download response from the host to the terminal, where ␣ denotes a blank space.

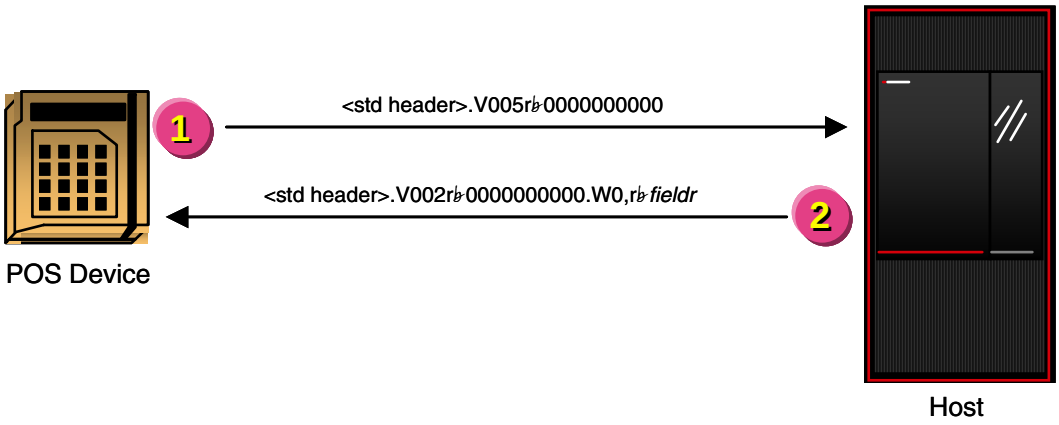
```
.V005r␣0000000000.W0,r␣fieldr
```

The key describing the information contained in the example is as follows:

.	=	Field separator
V	=	FID V
00	=	Category code (download for a DID completed)
5	=	Access code (5 indicates to read a specific download field)
r$\not b$	=	Processing flag (DID r)
0s	=	Filler
.	=	Field separator
W	=	FID W
0	=	Fill byte (always ASCII 0)
,	=	Group separator
r$\not b$	=	DID r (requested DID)
fieldr	=	Download data for DID r

Partial Download Example

The example shown below is of a partial download. The data shown is assumed to contain a standard header and be terminated with an ETX. The period (.) represents a field separator character, with a comma (,) representing a group separator character. Explanations of the messages are included below.



1. The terminal makes a download request that includes the download key field, indicating which download field is desired. In this example, DID r is requested.
2. The host uses the download key to indicate that this is a partial download. The host accesses the requested field and responds to the terminal with just that field and a response code of 880 in the header, indicating that no more data is to be downloaded.

4: Processing Considerations

Besides setting up ACI standard POS message formats, customers and vendors are responsible for making decisions concerning several additional configuration issues. These issues determine the way ACI standard POS terminals and the host process transactions. This section discusses the following configuration processing issues the customer and vendor must consider when using the ACI standard POS message:

- Configurable receipts
- Returning account balances
- Chargebacks for preauthorization completions
- Draft capture
- Message sequencing
- Transaction accumulation totals
- PIN encryption
- Europay, MasterCard, and Visa (EMV) Transaction Certificates
- Data encryption
- Message authentication codes (MACs)
- American Express card security codes (CSCs)
- Derived unique key per transaction (DUKPT)
- Dynamic key management
- Handshaking
- Mail support
- Interac Online Payment transaction identification requirements

Configurable Receipts

The host does not format receipts. Instead, it is the customer's responsibility to configure each response in the terminal configuration data to include sufficient data for the terminal to format and print a receipt.

However, the host has the capability of returning a response determined by the customer in up to three different languages as well as returning a 48-character response.

This subsection provides information on the following topics concerning receipts:

- Receipt information
- Language for responses
- Terminal responses

Receipt Information

Although it is the customer's responsibility to determine the information to be included on receipts, several suggestions follow.

Standard Message Header Fields

The standard header fields typically printed on a receipt include the following:

- Terminal ID
- Employee ID
- Date
- Time
- Transaction code
- Response code

Optional Data Fields

The optional data fields typically printed on a receipt include the following:

FID	FID Name
B	Amount 1
D	Application Account Type
E	Application Account Number
F	Approval Code
J	Available Balance
K	Business Date
h	Sequence Number
k	Terminal Location
q*	Track 2/Customer
r*	Track 2/Supervisor
s	Transaction Description
2*	Track 1/Customer
3*	Track 1/Supervisor

* Either Track 2 or Track 1 data must be specified in the host configuration to print the primary account number (PAN).

Response Language

Although the customer and the terminal vendor are responsible for determining what the receipt looks like, several aspects of the receipts must be set up through the host software. These include setting parameters so the host knows which response to return to the terminal and developing the exact text of the responses to be sent.

Language Code

The host uses the request first to determine the language in which to return responses. Specifically, the host checks for the presence of the language code field identified with field identifier (FID) U. If present, the host checks the value in this field. If this field is not included in the request, the host uses the default language code set in the host database.

Therefore, the customer is required to set a default language code in the host database. If a specific language code is required for certain transactions or for any other reason, the customer needs to include FID U in the request message. For consistency, the host configuration should also be set appropriately, indicating when FID U is to be included in requests.

Language Index and Responses

The host uses the language code, along with the response code returned from the authorizer, to determine the response text to send to the terminal.

Terminal Responses

If the host is set to include the Response Display field (FID g) in responses to the terminal, the host returns a customer-configured response to the terminal. Although these responses are typically used to inform the terminal operator of processing results, the response text is optional and up to the discretion of the customer. The only restriction on these responses is that they not exceed 48 characters. If the response is less than 48 characters, trailing spaces are removed by the host.

These customer-configured responses are based on transaction type. For example, different responses can be made for each transaction type. Up to the maximum field length, one or more of the optional or download data fields shown on the following page can be included in any display.

FID	FID Name
B	Amount 1
C	Amount 2
E	Account number
F	Approval code
J	Amount 1 (on balance inquiry transactions)
K	Terminal business date
N	Customer ID
Q	Echo data
S	Invoice number
T	Original invoice number
Z	Address verification status code
d	Retailer ID
h	Sequence number
k	Terminal location

Returning Account Balances

The host returns account balances on balance inquiry transactions. In addition, the host can be configured to return balances on purchase transactions and other transactions. The option for the host to return balances on purchase transactions is set at the card prefix level, while the option for the host to return balances on transactions other than balance inquiries and purchases is set at the terminal level.

The host returns the available balance in FID J in response messages.

The host can encrypt the account balance. Refer to the Data Encryption discussion in this section for more information.

The host can be configured to reverse balance inquiry transactions if they do not complete as intended. While balance inquiry transactions typically are not reversed, it may be appropriate to do so when a service charge is applied for an inquiry. This configuration option applies only to the balance inquiry transaction, not the return of balances on other transactions.

Chargebacks for Preauthorized Hold Completions

A preauthorized hold transaction requests that funds be placed on hold against a future purchase or payment. These funds are not available to the cardholder until the preauthorized hold request expires or the host system receives a preauthorized completion transaction indicating that the purchase or payment for which funds were being held was completed as authorized. The host supports standard and enhanced preauthorized holds processing.

The host cannot reject a preauthorized completion transaction. However, it can be configured to generate a chargeback (0402 message) following a preauthorization completion transaction that does not meet the approval requirements.

- In standard preauthorized holds processing, a chargeback can be generated if the hold has expired.
- In enhanced preauthorized holds processing, a chargeback can be generated if any of the following processing requirements are not met.
  - The preauthorization completion amount must not exceed the preauthorization hold amount.
  - The preauthorization hold must not have expired.
  - The preauthorization hold completion timer must not have expired.

Draft Capture

The ACI standard POS message allows for processing retailer transactions as draft capture or non-draft capture. Non-draft capture transactions are transactions for which the paper draft represents the actual settlement instrument. Paper drafts must be physically presented for deposit in order for the transaction to be settled. While the transaction may have been authorized by the host application, the host record cannot be used to settle the item.

Draft capture transactions are electronically captured at the time of the transaction and are sufficient in themselves to enact a funds transfer for the transaction amount. The host identifies draft capture transactions in its files and includes draft capture totals on its retailer reports so that the retailer sponsor can settle with the retailer.

Draft capture is a settlement tool only. It has no impact on the authorization of transactions. Cardholders are still checked against the same authorization criteria without regard for whether a transaction is draft capture or non-draft capture. Thus, the retailer is not incurring additional risk of cardholder fraud by using or not using draft capture.

Draft Capture Options

Customers using the ACI standard POS device message specifications have three draft capture options available to them. These are as follows:

- Authorization only with paper draft follow-up
- Authorize and draft capture
- Terminal-defined draft capture

Different draft capture options can be selected for each terminal and card type. Customers indicate the draft capture option they have selected for each terminal and card type in the host database. This data is downloaded to terminals in the Transaction Profile field of download field identifier (DID k). The valid values for the Transaction Profile field are as follows:

0 = Authorize only with paper follow-up
1 = Authorize and draft capture
2 = Authorize and expect electronic follow-up
3 = Terminal determines data capture mode for each transaction

For request or force-post transactions from a POS terminal, the transaction profile from the terminal can be 0 or 1. For a reversal transaction from a POS terminal, the transaction profile must be the same as that of the original transaction.

Although the value of 2 (authorize and expect electronic follow-up) is generally acceptable in this field, some host applications may not support the value. The value for the terminal in the host database is used, unless the value in the host database indicates the terminal is to determine the draft capture value.

If the terminal is to determine the draft capture value, the host uses the value sent in FID P (Draft Capture Flag) of the request. FID P contains the following values for the host:

0 = Authorize only with paper follow-up
1 = Authorize and draft capture

Authorization Only With Paper Follow-up

The authorization only with paper follow-up method of draft capture obtains online authorization for the amount of the customer purchase or other type of data, depending on the transaction. In an authorization only environment, the card issuer waits for paper follow-up before posting the transaction and settling with the retailer. If a card type is set up to use this option, none of the transactions for the card type from the specified terminal are draft capture.

With this method, the host updates the cardholder usage and activity totals, logs transactions, and updates the service totals and authorization totals for the terminal in the host database.

The host waits for paper to post and settle with retailer and on-us cardholder accounts.

The value in the host database, or the value in FID P, must be set to 0 or 3 with the terminal sending 0, indicating to authorize with paper follow-up.

Authorization and Draft Capture

The authorization and draft capture option is a one-step process that allows settlement to occur without paper follow-up. This option performs an authorization on the transaction and at the same time automatically captures the draft electronically. Using electronic draft capture, no paper follow-up is required by the card issuer to post the transaction and settle with the retailer. The procedure is completed in one transaction.

With this method, the host updates the cardholder usage and activity totals, logs transactions, and updates service totals, authorization totals, and draft capture totals for the terminal in the host database.

The host posts and settles with the retailer and on-us cardholder accounts without paper follow-up.

The value for the terminal in the host database, or the FID P (Draft Capture Flag), must be set to 1 or 3 with the terminal sending 1, indicating to use the authorization and draft capture method.

Terminal-Defined Draft Capture

The terminal-defined draft capture option allows the terminal to determine the draft capture mode for each transaction based on card type. If a card type is set up with this option, the terminal selects one of the draft capture options described previously (i.e., authorization only with paper follow-up or authorization and draft capture) for each transaction involving the card type. The host uses the value sent in FID P (Draft Capture Flag) of the request, as described previously.

Message Sequencing

An ACI standard POS-compliant terminal can request the host to validate transmission numbers and sequence numbers by including these in requests. Transmission number checking is only intended to avoid duplicates. Sequence number checking is intended to ensure that the host has received and processed every transaction only once. Any combination of these checking techniques can be implemented.

Transmission Number Checking

Transmission numbers can be included in the Transmission Number field in the standard message header. If no transmission number checking is required by the customer, the Transmission Number field in the standard message header must be set to zeros. If the customer wants to check transmission numbers, the Transmission Number field in the standard message header must be set to a nonzero value. Any nonzero values in the Transmission Number field are checked by the host. Although the use of the transmission number field is optional, ACI recommends that most POS terminals use it, especially when terminals have offline authorization capabilities and send transactions in a store-and-forward mode.

For online transactions (Message Subtype field in the standard message header set to a value of O), it is theoretically sufficient for the terminal to supply only the sequence number on requests as a means of controlling duplicate transmissions. When the host receives an online request with a sequence number that it is not expecting, it can execute the resynchronization procedure described later in this section, thereby avoiding duplicate processing.

If the transmission number is not included as recommended, the host cannot guarantee adequate protection against posting transactions twice in situations where the terminal times out or when processing force-post transactions.

Transmission numbers must be used in the manner specified below.

- The terminal must assign a new transmission number to each new transaction. The host expects the terminal to increment the number from 1 to 99 and then roll back to 1 again.
- When a new terminal record is initialized at the host, the transmission number is set to 00, so the terminal can be configured to send 01 as an initial value.

- When transactions are retransmitted as force-post transactions due to a terminal timing out, the transmission number that accompanied the original transaction is used on the force post transaction.
- When the host receives a message having the same transmission number as the last message it received, and the last message received by the host was approved, the message is declined with a response code of 078, indicating the transaction is a duplicate. When the host receives a message having the same transmission number as the last message received by the host, and the last message received by the host was declined, the message is processed.

Store-and-Forward Considerations

For store-and-forward transactions (Message Subtype field in the standard message header set to a value of S), no message resynchronization processing is done. The transaction is processed without checking its sequence number.

With store-and-forward transactions, the possibility exists for the terminal to transmit the transaction, but not receive a response from the host. The terminal has no way of determining whether the host actually received the transaction and has no alternative but to retransmit until a valid response is received.

Because the host does not support sequence number resynchronization for store-and-forward transactions, it processes any retransmission as a new transaction unless there is some way to determine that the transaction has already been processed.

The only way to do this check on store-and-forward transactions is through the use of transmission numbers. The terminal should assign the transmission number to a transaction when it is sent for the first time. Only after a valid response for the transaction is received from the host should the terminal increment the transmission number for use on the next transaction. This way a terminal can retransmit repeatedly knowing that the host only processes the transaction once, regardless of communication line failures.

Force-Post Considerations

The host offers an alternate solution to the duplicate detection problem. Transactions that are authorized offline by the terminal can be sent as force-post transactions (Message Subtype field in the standard message header set to a value of F). In this case, sequence number checking logic is applied, exactly as with online transactions, before force posting the transaction.

This alternate solution is less efficient than the transmission number solution because it compromises the distinction that the host draws between force-post and store-and-forward transactions.

A force-post transaction is technically a transaction for which prior authorization was received. For example, a preauthorization purchase completion transaction is considered a force-post transaction because it received prior authorization with a preauthorization purchase transaction. Another example of a force-post transaction is a transaction for which the clerk received authorization by calling a referral authorization call center.

A store-and-forward transaction is one for which the terminal is standing in without communicating with an outside authorizer. In some cases, if offline-authorized transactions are sent to the host as force-post transactions, they may not be identified as having been authorized by a stand-in POS terminal on host application reports.

Sequence Number Checking

Sequence number checking is intended to ensure that the host has received and processed each transaction only once. When the host encounters the Sequence Number field (FID h) in a request, it verifies that the sequence number in this field is the one the host is expecting. The host checks the sequence number sent in the request with the sequence number stored for the terminal in the host database.

Sequence number checking is done by the host only when FID h is included in the request from the terminal. This field consists of 10 numeric characters. The first three characters indicate the shift number and range from 001 to 999, rolling back to 001. The next three characters identify the current batch and range from 001 to 999, rolling back to 001. The next three characters consist of a unique sequence number within the shift and batch and ranges from 001 to 999, rolling back to 001. The last character is a reset flag, indicating whether the terminal or the host is responsible for determining the correct sequence number.

The host is initialized to expect sequence number 001001001, with the shift number set to 001, the batch number set to 001, and the sequence number set to 001. From this point on, the host expects the sequence number to increment by one. When 999 is reached, the host expects 001 as the next sequence number.

Batch Close Transactions

When the first batch close transaction is performed, the host next expects a sequence number of 001002001. Since the batch close transaction is optional, the host allows the terminal to send a sequence number with the sequence number set to 001 without incrementing the batch number to 002. When this occurs, the host can perform an implicit batch close.

Shift Close Transactions

When the first shift close transaction is performed, the host next expects a sequence number of 002001001. Since the shift close transaction is optional, the host allows the terminal to send a sequence number with the batch number set to 001 without incrementing the shift number to 002. When this occurs, the host can perform an implicit shift close transaction. An implicit batch close transaction can also be performed at this time.

Close Day Transactions

When the first day close transaction is performed, the host next expects a sequence number of 001001001. Since the day close transaction is optional, the host allows the terminal to send a sequence number with the shift number set to 001 without resetting the shift, batch, and sequence number to 001. When this occurs, the host can perform an implicit day close transaction. An implicit shift close transaction and an implicit batch close transaction can also be performed at this time.

Implied Closes from the Terminal

Implied closes are processed only if the terminal group is configured to support implied closes for the terminal in the host database. If the terminal group is not configured to support implied closes, the host does not perform closes and the host reporting programs do not execute correctly. In addition, implied closes are completed only if the reset flag in the sequence number is set to 1, indicating that the terminal is to determine the sequence number.

Unexpected Sequence Number, Batch Number, or Shift Number

When the host receives an unexpected sequence number, batch number, or shift number and the reset flag in the Sequence Number field (FID h) is set to 0, indicating that the host is to determine the sequence number, the host responds to the terminal with a denial response code of 899 and FID h. Response code 899 indicates a sequence error has occurred and resynchronization is to be performed. FID h contains the expected sequence number.

The terminal has two options when it receives a response code of 899 with FID h. It can adjust its sequence number, batch number, or shift number to correspond with the sequence number, batch number, or shift number received from the host. This implies that the terminal is able to adjust its sequence numbers, batch numbers, and shift numbers backward or forward in its internal queue and send subsequent transactions that correspond to the adjusted sequence number, batch number, or shift number.

The terminal can also direct the host to adjust its sequence number, batch number, or shift number to match the sequence number, batch number, or shift number maintained by the terminal. To do this, the terminal needs to put the sequence number, batch number, and shift number that the terminal determines to be correct in FID h, set the reset flag to 1, and send the transaction that originally received a response code of 899 back to the host. Once the transaction is resubmitted to the host, the host accepts the transaction.

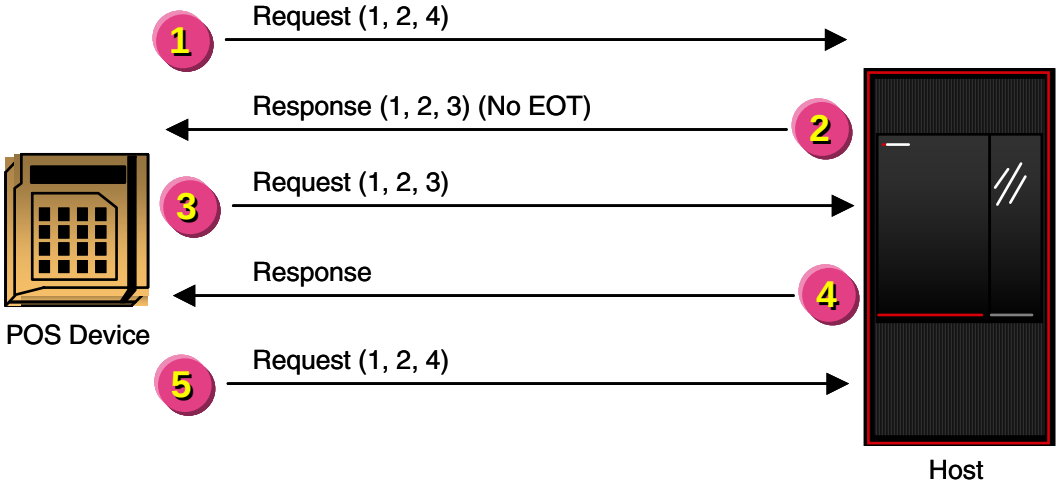
In addition, if the batch or shift number has been changed from the batch or shift number in the original transaction, the host determines that the transaction being resubmitted contains a new batch or shift number and the host performs an implied batch close or an implied shift close. The host then processes the transaction and responds to the terminal.

Sequence Number Checking Examples

The examples on the following pages show the processing that occurs between the terminal and the host during various sequence number checking scenarios. Each scenario includes a graphic illustrating the processing flows between the terminal and the host, and each graphic is followed by descriptions of each step exchanged in the flow.

Terminal Backward-Adjusted Sequence Number

The following example describes a situation where the host is expecting a sequence number other than the one requested by the terminal. This example is applicable only in situations where the terminal is able to search backwards in its transaction queue to locate a sequence number.

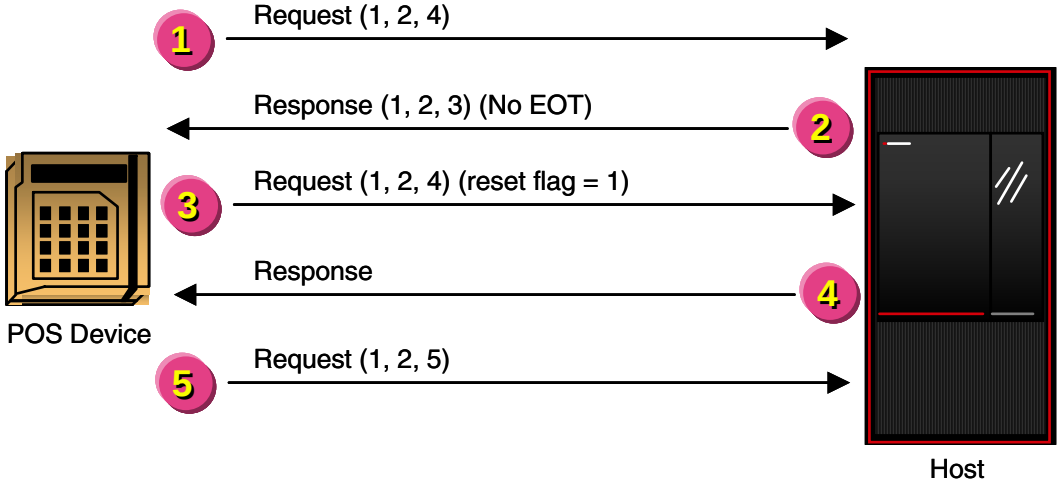


1. The terminal generates a transaction request to the host, and includes the Sequence Number field (FID h) with a value of 1, 2, 4 (shift 1, batch 2, sequence number 4 within current batch).
2. For example purposes, it is assumed that the host is expecting 1, 2, 3. The host responds to the terminal request with response code 899 (sequence error–resync), and includes FID h to indicate a sequence number of 1, 2, 3.

   Note: The host does not issue a disconnect command with response code 899.
3. The terminal receives the response from the host and searches its internal transaction queue until it locates sequence number 1, 2, 3. The terminal then sends the transaction with sequence number 1, 2, 3 to the host.
4. The host processes the transaction with sequence number 1, 2, 3 and responds to the terminal.
5. The terminal then sends the next transaction with sequence number 1, 2, 4, which the host is now expecting.

Terminal Forward-Adjusted Sequence Number

The following example describes a situation where the host is expecting a sequence number other than the one requested by the terminal. This example is applicable only in situations where the terminal is able to search forward in its transaction queue to locate a sequence number.

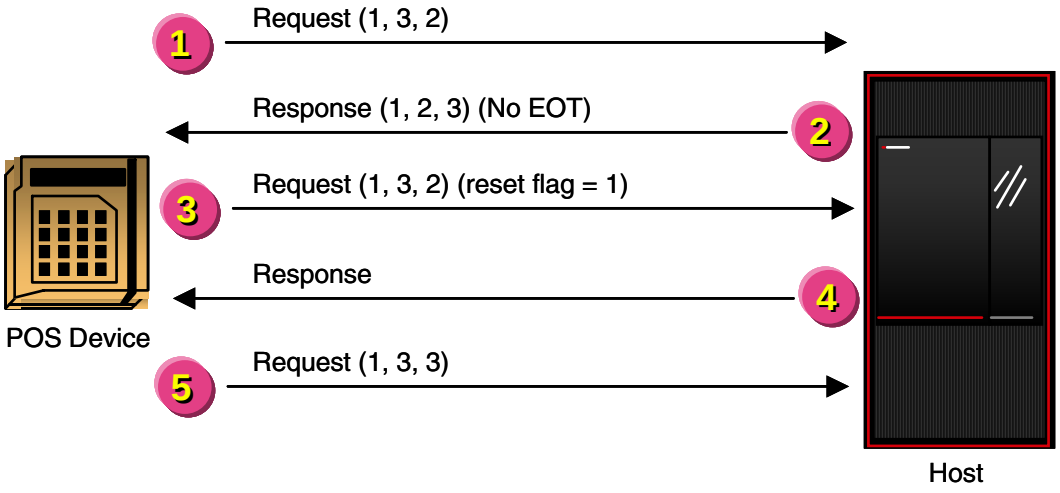


1. The terminal generates a transaction request to the host, and includes the Sequence Number field (FID h) with a value of 1, 2, 4 (shift 1, batch 2, sequence number 4 within current batch).
2. For example purposes, it is assumed that the host is expecting 1, 2, 3. The host responds to the terminal request with response code 899 (sequence error–resync), and includes FID h to indicate a sequence number of 1, 2, 3.

   Note: The host does not issue a disconnect command with response code 899.
3. The terminal is unable to search backwards in its internal transaction queue to locate sequence number 1, 2, 3. Thus, the terminal resubmits the transaction to the host with sequence number 1, 2, 4 and the Reset Flag set to a value of 1.
4. The host accepts and processes the transaction with sequence number 1, 2, 4 and responds to the terminal.
5. The terminal then sends the next transaction with sequence number 1, 2, 5, which the host is now expecting.

Terminal Forward-Adjusted Sequence Number (Implicit Batch Close)

The following example describes a situation where the host is expecting a sequence number other than the one requested by the terminal. This example is applicable only in situations where the terminal is able to search forward in its transaction queue to locate a sequence number.



1. The terminal generates a transaction request to the host, and includes the Sequence Number field (FID h) with a value of 1, 3, 2 (shift 1, batch 3, sequence number 2 within current batch).
2. For example purposes, it is assumed that the host is expecting 1, 2, 3. The host responds to the terminal request with response code 899 (sequence error–resync), and includes FID h to indicate a sequence number of 1, 2, 3.

   Note: The host does not issue a disconnect command with response code 899.
3. The terminal is unable to search backwards in its internal transaction queue to locate sequence number 1, 2, 3. For example purposes, it is assumed that the terminal is unable to search backwards any further than 1, 3, 2. The terminal thus resubmits the transaction with a sequence number of 1, 3, 2 with the Reset Flag set to 1.
4. The host accepts the transaction with sequence number 1, 3, 2. At this point, the host determines that this is a new batch and performs an implied batch close. The host then processes the transaction and sends a response to the terminal.

5. The terminal then sends the next transaction with sequence number 1, 3, 3, which the host is now expecting.

Resetting the Sequence Number

A parameter determines how the host resets the sequence number when a Day Close transaction is performed. The following options are available:

- Use standard reset procedures (reset, shift, batch, and seq # values)
- Reset shift and seq # values, but increment batch value
- Reset batch and seq # values, but increment shift value
- Reset seq # value, but increment shift and batch values
- Increment shift, batch, and seq # values

The standard sequence number reset procedure is resetting shift, batch, and sequence number (seq #) values to 001 when a Day Close transaction is performed. Under most operating conditions, this procedure is sufficient to provide unique sequence numbers for all transactions. However, in certain environments where a cardholder does not use ATMs and uses the same POS terminal on a regular schedule, the possibility exists for transactions performed on different days to have the same sequence number. In this situation, the host could decline a transaction because the transaction sequence numbers are the same, even though the transactions are unique because they are performed on different days.

The host can be configured to increment various shift, batch, and sequence number (seq #) combinations as a way to eliminate such duplicate sequence number situations from occurring, as shown in the following table. Values that differ from the standard reset are ***highlighted in red italics.***

Transaction	Standard Reset	Reset Shift and Seq #, Increment Batch	Reset Batch and Seq #, Increment Shift	Reset Seq #, Increment Shift and Batch	Increment Shift, Batch, and Seq #
Tran #1 of day	001001001	001001001	001001001	001001001	001001001
Tran #2 of day	001001002	001001002	001001002	001001002	001001002
Tran #3 of day	001001003	001001003	001001003	001001003	001001003

Transaction	Standard Reset	Reset Shift and Seq #, Increment Batch	Reset Batch and Seq #, Increment Shift	Reset Seq #, Increment Shift and Batch	Increment Shift, Batch, and Seq #
Batch close	001002000	001002000	001002000	001002000	001002003
Tran #4 of day	001002001	001002001	001002001	001002001	001002004
Tran #5 of day	001002002	001002002	001002002	001002002	001002005
Tran #6 of day	001002003	001002003	001002003	001002003	001002006
Batch close	001003000	001003000	001003000	001003000	001003006
Tran #7 of day	001003001	001003001	001003001	001003001	001003007
Batch close	001004000	001004000	001004000	001004000	001004007
Shift close	002001000	002004000	002001000	002004000	002004007
Tran #8 of day	002001001	002004001	002001001	002004001	002004008
Tran #9 of day	002001002	002004002	002001002	002004002	002004009
Batch close	002002000	002005000	002002000	002005000	002005009
Shift close	003001000	003005000	003001000	003005000	003005009
Day close	001001000	001005000	003001000	003005000	003005009
Tran #1 of next day	001001001	001005001	003001001	003005001	003005010

Additional Store-and-Forward Considerations

When considering how to implement message sequencing, the processing of store-and-forward transactions must be considered. The interspersing of online transactions with store-and-forward transactions subverts duplicate message detection logic in the host. ACI recommends that manufacturers design their terminals to process store-and-forward transactions in a first-in, first-out (FIFO) sequence. As new transactions occur at the terminal, they should be added to the end of the stored transactions so that in all cases the oldest transactions are sent to the host first.

In addition, ACI recommends assigning the sequence number as the transaction is initialized and the transmission number as the transaction is sent to the host.

Transaction Accumulation Totals

The customer can choose to have the terminal accumulate any set of batch, shift, and day totals and forward them to the host with the corresponding close request. If the totals differ from host totals, they are recorded in a transaction log file. No attempt is made to indicate or reconcile any differences.

The host maintains totals for each terminal at a merchant site. Additionally, the host can maintain a set of totals that combines all of the terminals at a merchant site. Terminals in a department store or a pay-at-the-pump service station are examples of site configurations.

The host always accumulates batch, shift, and day totals. When an explicit or implicit close for the corresponding period is received, the host records these totals in the transaction log file and resets the totals as appropriate. The host also supports reconciliation with a series of subtotals request transactions prior to an explicit or implicit close.

A flag for the terminal in the host database indicates to the host which totals are supported by the terminal by means of close transactions. Totals for the terminal in the host database are allowed to accumulate until reset by means of a close transaction.

The host records clerk totals as required by the terminal. It logs these totals to the transaction log file. When a clerk totals request is received from the terminal, the host accumulates the clerk totals by employee ID or terminal ID and returns them to the terminal.

PIN Encryption

Customers are not required to support PIN encryption, but if they do decide to support it, the host offers several methods of PIN encryption. It is up to the customer to determine which method best suits their particular requirements.

Depending on how the terminal is configured in the host database, the PIN/Customer field (FID b) and the PIN/Supervisor field (FID c) can be encrypted using the following methods:

- Single-length key (16 bytes), single DES performed in software
- Single-length key (16 bytes), single DES performed in hardware
- Double-length key (32 bytes), triple DES performed in hardware
- Triple-length key (48 bytes), triple DES performed in hardware

If PIN encryption is selected, the terminal must support the manual entry or injection of a master key or base derivation key. Master/session keys are used with master/session key management. Base derivation keys are used when the POS devices derive a unique key per transaction (DUKPT).

Master/Session Key Management

When using master/session key management, the host uses data in the host database to determine when to provide the terminal with a new communications (session) key. The host generates a new communications key, which is also known as the KPE, under the following conditions:

- When a response message is configured to contain FID M (Communications Key).
- When a dynamic key management threshold is reached. Refer to the Dynamic Key Management discussion in this section for more information.
- During a download if download field identifier (DID) g (Communications Key) is to be sent.

If the host is required to perform PIN decryption and the communications key is all spaces, the host assumes the terminal has never received a communications key from the host and rejects the PIN as invalid.

Europay, MasterCard and Visa (EMV) Transaction Certificates

When an EMV transaction is authorized offline by the terminal or EMV card, the card generates a Transaction Certificate (TC), which is a card-generated cryptogram containing information about the transaction that can be used if the transaction is disputed.

The host can accept TCs uploaded, one at a time, from the terminal. Each TC is sent to the host as an EMV log-only transaction identified by the message subtype E in the message header. The TC is carried in the Application Cryptogram (AC) field in FID 6 subFID O (EMV Request Data).

If the terminal does not receive a response to an EMV log-only transaction request prior to timing out, the terminal should immediately send an EMV log-only cancellation request (message subtype V). From the perspective of the terminal, an EMV log-only cancellation should be handled similarly to a timeout reversal (message subtype T):

- The EMV log-only cancellation must immediately follow the original EMV log-only transaction.
- The EMV log-only cancellation must take precedence in the queue over online requests and SAF requests.
- The transmission number of the EMV log-only cancellation request must be set to match that of the original EMV log-only transaction. The transmission number is used to match the EMV log-only cancellation with the original EMV log-only transaction.

If an EMV log-only cancellation request is declined, a Response Code value of 050 (general decline) is returned in the message header.

Data Encryption

The SPDH data encryption module is an additional licensable module that, when bound with the standard SPDH module, supports data encryption for FID J (Available Balance). The SPDH also supports full message encryption and configurable message encryption. DUKPT is supported with data encryption.

Before encrypted data other than PINs and MACs can be returned to a terminal, the data encryption terminal master key must be manually entered or injected into the terminal. The same key also must be entered in the host database. The other key used with data encryption is the data encryption communications key, a working key that is also known as the message encryption key (KME). The data encryption key is encrypted under the data encryption terminal master key before it is downloaded to the terminal.

The host generates a new data encryption communications key under the following conditions:

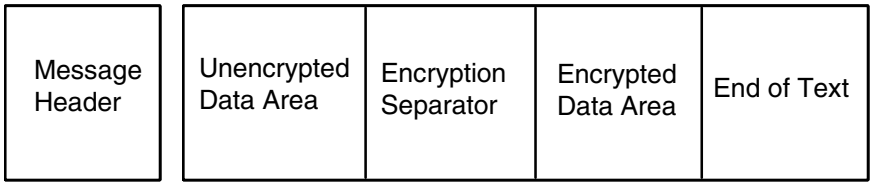
- When a response message is configured to contain FID I (Data Encryption Key).
- When a dynamic key management threshold is reached. Refer to the Dynamic Key Management discussion in this section for more information.
- During a download if download field identifier (DID) j (Data Encryption Key) is to be sent.

The ACI standard POS message supports full message encryption and configurable message encryption. Full message encryption allows the institution to encrypt all optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. Configurable message encryption allows the institution to encrypt specific optional data fields, except G, H, I, M, b, and c, in the ACI standard POS message. The optional data fields are configured to be encrypted in the ACI standard device configuration file request and response field maps. Full message encryption and configurable message encryption are enabled using the data encryption field in the POS terminal data file.

For DUKPT data encryption, optional field 6 subfield T (Key Serial Number and Descriptor) must not be encrypted, but the remaining subfields can be encrypted or unencrypted. To do this, field 6 appears in the message twice. For example, the first field 6 would contain subfield T and would reside in the unencrypted data area of the message. It could also contain other subfields that are unencrypted and would reside in the encrypted data area of the message after the encryption separator.

Whether using full message encryption or configurable message encryption, the format of the ACI Standard POS Message is the same. The message begins with the standard message header, which is never encrypted. Following the standard message header is the unencrypted data area, which contains all the optional data fields and subfields that are not encrypted under the data encryption key. Next comes the encryption separator (hexadecimal 1F), a separator character that is used to mark the beginning of the encrypted data area. The encrypted data area extends from the encryption separator to the end of the message and contains all of the optional data fields and subfields that are encrypted under the data encryption key.

Field separator characters (hexadecimal 1C) and field identifiers (FIDs), which preface optional data fields, are included in the encrypted data. Likewise, record separator characters (hexadecimal 1E) and subfield identifiers (subFIDs), which preface optional data subfields, are included. The encryption separator itself is not encrypted, nor is the end-of-text character at the end of the message, if present. The following is an illustration of the format of an encrypted ACI standard POS message.



By grouping optional data fields and subfields into the encrypted data area, the SPDH is able to encrypt or decrypt the data with a single call to the host security module (HSM).

Message Authentication

The host support of message authentication codes (MACs) uses the standards documented in the ANSI X9.19 (1986) standard, ***Financial Institution Retail Message Authentication Standard***. MACs, along with the ANSI standard, are designed to protect financial transaction messages against accidental or deliberate alteration. In addition, they protect against the fraudulent insertion of messages.

The customer determines whether to use MACs and, if so, how they are implemented. MAC generation and verification options include the following:

- Single-length key (16 bytes), single DES performed in software
- Single-length key (16 bytes), single DES performed in hardware
- Double-length key (32 bytes), triple DES performed in hardware
- Triple-length key (48 bytes), triple DES performed in hardware

If MACs are used in any form, the following three fields in the standard message header are authenticated by the host in every request:

- Transmission Number
- Terminal ID
- Transaction Code

The same three fields are authenticated by the terminal in every response using MACs in any form. In addition, the Response Code field in the standard message header is always authenticated in responses using MACs.

Any number of optional fields totaling up to 1000 bytes can be verified using MACs. Additional information about each of these fields can be found in section 2.

In addition, one or more of the following working keys can be included when MACs are computed.

- The PIN communications key (KPE), which is returned in FID M.
- The MAC communications key (KMAC), which is returned in FID H.
- The data encryption communications key (KME), which is returned in FID I.

Setting Up MACs

Any number of additional optional data fields can be verified using MACs, although the host does not support authenticating the entire message. While the data within the fields are verified using MACs, the FIDs identifying the optional data fields are not included in the MAC.

The fields to be verified using MACs are set in the host configuration for requests and responses for every transaction type. However, the customer and the vendor must agree on the fields using MACs in order for the messages to be verified.

ACI recommends that several optional data fields be included when MACS are used. These fields, including their field identifier (FID), are as follows.

Amount 1 (FID B)
Amount 2 (FID C)
Approval Code (FID F)
PIN/Customer (FID b)
Track 2/Customer (FID q)
Track 1/Customer (FID 2)

Whenever MACs are used at a terminal that conforms to the host, the Authentication Code field (FID G) must be included in both the request and the response. In addition, the MAC terminal master key must be manually entered or injected into the terminal. The same key also must be entered in the terminal data in the host database. The other key used with MACs, the MAC communications key (KMAC), is randomly generated by the host.

Generating a New MAC Communications Key

The host generates a new MAC communications key under the following three conditions:

- When a response message is configured to contain FID H (Authentication Key).
- When a dynamic key management threshold is reached. Refer to the Dynamic Key Management discussion in this section for more information.
- During a download if download field identifier (DID) h (Authentication Key), is to be sent.

Whenever a new MAC communications key is sent to the terminal in a response, it is encrypted under the MAC terminal master key. Once a new key is generated, the host updates the terminal data in the host database.

Failed MAC Procedure

When the host receives a request that cannot be verified using MACs, it formats a response message with a response code of 898, indicating an invalid MAC request. The host also increments the invalid MAC counter for the terminal in the host database.

When a terminal is unable to verify a MAC received from the host in a transaction response with monetary impact, the terminal must generate a reversal. The reversal is identical to the response message, with two exceptions. The response code is set to 989, indicating an invalid MAC response. Also, the Message Subtype field in the standard message header is set to a value of R, indicating a reversal. The reversal is not verified using a MAC.

When the host receives a MAC reversal from a terminal, it sends a reversal message to the transaction authorizer. This MAC reversal must be the next request the authorizer receives from the terminal. If any other requests from the terminal are processed before the reversal, the reversal is dropped.

Derived Unique Key Per Transaction

POS devices attached to the host can use derived unique key per transaction (DUKPT) key management for PIN communications (session) keys, MAC communications keys, and message encryption keys.

When using derived key per transaction (DUKPT) key management, a unique base derivation key must be manually entered or injected into each POS device. The same base derivation key must be loaded into a database maintained by the host. When a POS device derives a unique key for a transaction, it must include the Key Serial Number and Descriptor field (FID 6, subFID T) in the request message sent to the host.

American Express Card Security Codes (CSCs)

The ACI standard POS message contains data enabling hosts to verify American Express card security codes. The three types of card security codes (CSCs) are as follows:

- Three-digit CSC located on the signature panel
- Four-digit CSC located on the front of the card
- Five-digit CSC located on the magnetic stripe

Card-swipe transactions use the CSCs as follows:

- The five-digit CSC is mandatory and is submitted as part of the Track 2 or Track 1 data. If this CSC is missing or is incorrect, the host rejects the transaction, regardless of whether a shorter CSC is also present.
- The four- and three-digit CSCs are optional.
- The four- and three-digit CSCs are mutually exclusive.
- If a transaction contains a five-digit CSC and one of the shorter CSCs, the host checks both CSC values in one call to the hardware security module. However, the shorter CSC verification is considered only when the five-digit CSC is correct.

Manually entered transactions use the CSCs as follows:

- The five-digit CSC cannot be submitted.
- The four- and three-digit CSCs are optional, meaning some transactions will not have either of these CSCs.
- The four- and three-digit CSCs are mutually exclusive.

Dynamic Key Management

The host, when configured with an additional licensable module, can support dynamic key management, which is the automatic replacement of working keys for a terminal based on one or more thresholds, for the following working keys:

- The PIN communications key (KPE), which is returned in FID M.
- The MAC communications key (KMAC), which is returned in FID H.
- The data encryption communications key (KME), which is returned in FID I.

The host uses thresholds based on several values in its database. If a threshold is configured with a nonzero value, the host generates a new key and sends it to the terminal with the next response whenever a threshold is reached. When a threshold is reached, the host includes the new key in the response regardless of the response configuration. Thresholds include the following:

- The total number of messages that can be authenticated with a MAC communications key.
- The number of failed message authentication attempts with a MAC communications key.
- The consecutive number of failed message authentication attempts with a MAC communications key. The host resets the counter associated with this threshold when a message is authenticated successfully.
- The total number of PINs that can be verified with a PIN communications key.
- The number of failed PIN verification attempts with a PIN communications key.
- The total number of times a data encryption communications key can be used.
- The number of times a data encryption communications key can be used unsuccessfully.

Handshaking

The customer must decide whether to support text-level handshake requests. The purpose of this request is to allow the terminal to verify the status of the communications link and, optionally, validate the communications key and authentication key for the terminal.

If the handshake request is made with only the standard message header, the host ignores every field in the header except for the Transaction Code, Processing Flag 1, and Processing Flag 2 fields. The host responds with the standard message header, including a response code of 007, and the valid local terminal date and time. In addition, appropriate values in the Processing Flag 1 and Processing Flag 2 fields can be sent in the response, if a mail message or a download is waiting. The host also returns any optional data fields configured by the customer.

If the handshake request includes the PIN/Customer field (FID b), the host decrypts the PIN and compares the decrypted PIN to 16 zeros. If they are not equal, the response code is set to 201, indicating an invalid PIN.

If the handshake request includes the Authentication Code field (FID G), the host validates the MAC, using the specified MAC FIDs. If the MAC in FID G does not equal the generated MAC, the response code is set to 898, indicating an invalid MAC.

Mail Support

If the host supports electronic mail, the customer must determine the degree to which electronic mail is supported by their POS terminals. For example, the customer can choose to allow a terminal only to receive mail, or the terminal can be allowed only to send mail. As another option, the customer can choose to support mail message transmission in both directions. A terminal can also be configured to accept unsolicited mail, or it can be configured to not allow mail to be sent to the terminal without the terminal first initiating a read mail transaction. In short, customers can choose from several options of sending and receiving mail electronically.

Unsolicited Mail

When the host receives an unsolicited mail message intended for a terminal, the host checks the terminal data in the host database to determine if the terminal can accept unsolicited mail. The host also checks if the terminal is a dial-up terminal.

The host sends unsolicited mail messages to the terminal if the terminal is not a dial-up terminal and the terminal data in the host database indicates the terminal can accept unsolicited mail. If both of these conditions are met, the host sends unsolicited mail messages to the terminal even if the terminal has not sent a read mail request to the host. The only time the host does not send unsolicited mail messages to the terminal when these two conditions are met is if the terminal has outstanding requests being processed. Unsolicited mail is sent to the terminal containing a standard message header that appears as though the host actually received a read mail request. The Mail/Download Key field (FID V) is also included in the message and is configured so the terminal can return it in a subsequent read mail request to retrieve the next piece of undelivered mail.

If mail exists for a terminal and the terminal is unable to accept the mail (i.e., the terminal is a dial-up terminal) or is not configured to support unsolicited mail, the host sets Processing Flag 1 in the standard message header to a value of 1 in each successive response to the terminal. This value indicates to the terminal that mail is waiting. The host continues to notify the terminal that mail exists in this manner until the terminal generates a read mail request.

In situations where the terminal is configured to support unsolicited mail, only one unsolicited delivery attempt per piece of mail is made by the host. When the terminal receives a message with Processing Flag 1 set to a value of 1, the terminal should send a read mail request to retrieve the mail messages that are waiting.

Send Mail Request

The terminal can send mail to a data processing center (DPC) using the send mail transaction request. In order to support send mail transactions, the terminal must be able to compose a piece of mail.

The only optional request field required by the host for this transaction is FID W. For send mail transactions, FID W contains the Mail Text field, which is composed of a destination DPC and a maximum of 449 characters of mail text.

Read Mail Request

The terminal can request any outstanding mail messages using the read mail transaction request. Typically, FID V, Mail Key, is required in read mail transaction requests other than initial requests. If the terminal does not include FID V in an initial read mail request, the host assumes the terminal is required to read the first of any mail of any category. Thus, the initial read mail request that the terminal generates for any mail can be made without FID V. However, all subsequent read mail requests should contain FID V. FID V contains the following four types of information:

Category Code — The category code is included whenever the terminal generates a generalized read request. The category code is a 2-position alphanumeric field. Category 99 is reserved for download alerts. The category code is a user-defined value.

Access Code — The type of access desired. The access code is a 1-position numeric field. The valid values for this field are as follows:

1 = Read first message. This value directs the host to read the first piece of any mail existing for the terminal.
2 = Read any next message. This value directs the host to retrieve the next piece of any mail existing for the terminal.
3 = Read first undelivered message. This value directs the host to retrieve the first piece of mail existing for the terminal that has not already been delivered.
4 = Read next undelivered message. This value directs the host to retrieve the next piece of mail existing for the terminal that has not already been delivered.
5 = Read specific message. This value directs the host to retrieve a specific piece of mail existing for the terminal. A valid Mail ID field value must be present in FID V to use this value.

Processing Flag — Reserved for future use.

Mail ID — The date and identifier for the message. The mail ID is a 10-position alphanumeric field. The first six bytes contain the date (YYMMDD) of the message. The last four bytes contain the mail message number. This field specifies the starting point for a read next request or the specific mail item for a read specific request.

Read Mail Response

The host responds to a read mail request by placing the mail message in FID W, if it is equal to or less than 440 bytes and fits in the field. If the mail message fits in FID W, the response code is set to 880, indicating that no more data exists.

If the mail message does not fit, the response code is set to 881, indicating that more data exists. The host saves the message context for the anticipated read next request. The terminal echoes the FID V from the previous response in read next requests.

When returning a read mail response, the host also translates the value in the Access Code field to the value it anticipates will follow.

An 880 response, indicating that no more data exists for a particular piece of mail, does not mean that no more mail exists for the terminal. In order to determine if no more mail exist for the terminal, the terminal must make another Read Next Request. No more mail is implied when the response code is 880 and no mail text is included in the response. In this case, FID W is not included in the response.

Mail Delivered Request

Terminals are allowed to mark mail as delivered so that subsequent read undelivered mail requests ignore the mail and move on to the next undelivered piece. In order to perform this function, the terminal must submit a mail delivered request containing the Mail Key field (FID V) of the read mail request (echoed in the read mail response). In this case, the Mail ID field in FID V identifies the mail to be marked as delivered and the Processing Flag field in FID V is used to mark the mail as delivered.

Terminals must also respond to unsolicited mail they receive with a mail delivered request. If a terminal receives unsolicited mail and does not respond with a mail delivered request, the host electronic mail application can mark the terminal down.

No more unsolicited mail is sent to the terminal until a subsequent mail request from the terminal is processed by the e-mail application. If a terminal is capable of supporting unsolicited mail, an unsolicited mail message should be acknowledged by a mail delivered request.

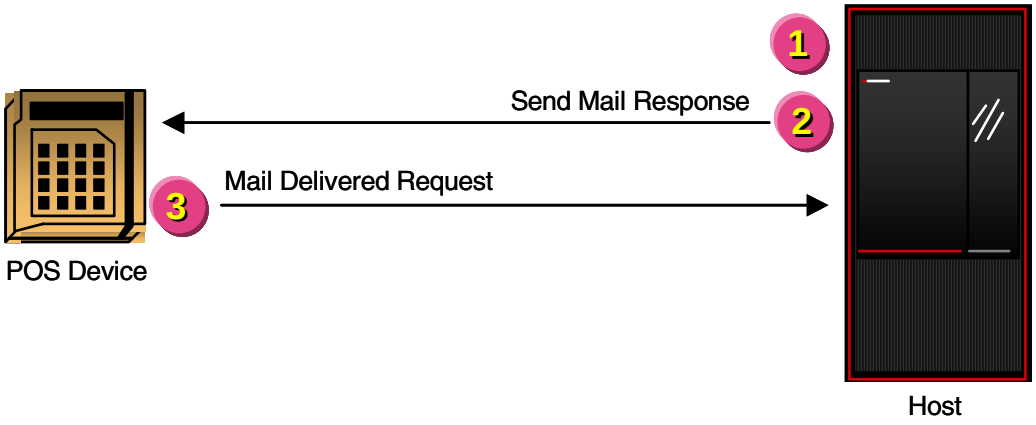
Message Flows

The following diagrams illustrate the exchange of mail between the host and the POS terminal during transaction processing. The data shown is assumed to follow the standard message header and to be terminated with an ETX in each situation. The quotes merely delimit the text and are not part of the message

Unsolicited Mail

The diagram below illustrates the transaction message flow in a scenario where the host receives unsolicited mail for a terminal.

Note: Dial-up terminals cannot receive unsolicited mail.



1. The host receives a message indicating that it must send a message to the terminal for immediate delivery.
2. The host determines whether the terminal can receive unsolicited mail as follows. For this example, assume that the terminal can receive unsolicited mail.
    a. If the terminal can receive unsolicited mail and there is no transaction in progress, the host formats a native mode message containing the mail message and sends it to the terminal. This message is formatted exactly

like a read mail response. FID V is also included and configured so the terminal can return it in a subsequent read mail request to retrieve the next piece of undelivered mail.

b. If the terminal cannot receive unsolicited mail or there is a transaction in progress, the host sets a flag for the terminal in the host database indicating that mail is waiting. The next time the host sends a response to the terminal, Processing Flag 1 in the standard header is set to a value of 1, indicating that mail is waiting.

3. The terminal sends a mail delivered request containing the Mail Key field (FID V) in the read mail response. In this case, the Mail ID field in FID V identifies the mail to be marked as delivered and the Processing Flag field in FID V is used to mark the mail as delivered for the terminal in the host database.

Send Mail Request

The diagram below illustrates the transaction message flow in a scenario where the terminal allows the operator to compose a piece of mail and route it to one of three DPCs.

1. The terminal allows the operator to compose a piece of mail and then formats a send mail request containing the mail text. This is included in the Mail/ Download Text field (FID W) along with the number of the DPC to which the mail is to be delivered. FID W is formatted as follows:

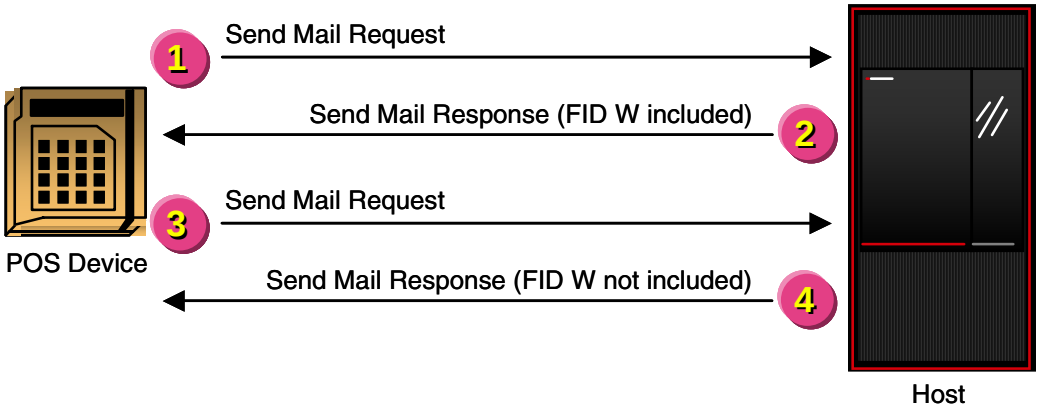
   `"`.W1*mail-text*`"`

2. The terminal sends the send mail request to the host.

3. The host either delivers the message to the request DPC or stores the message for future delivery if the DPC is unavailable. For the sake of this message flow, assume the mail is delivered successfully. The host then formats and sends a send mail response back to the terminal. In this response, the response code is set to a value of 870 (mail delivered).

Read Mail Request—Single Response

The following diagram illustrates a scenario where there is mail waiting to be delivered to the terminal and the terminal initiates a request to receive the mail. In this example, the text of the mail message is small enough for one response.



1. The terminal formats a read mail request and sends it to the host. In this scenario, the terminal requests the first of any undelivered mail using the Mail/Download Key field (FID V). FID V is formatted as follows:

   `"V00300000000000"`

2. The host retrieves the first piece of undelivered mail and formats a read mail response message. The entire mail message text fits within the allotted space of the Mail Text field (FID W) in the response message. Therefore, the response code is set to a value of 880 (mail message has been received in its entirety) and FID V is formatted as follows:

   `".V00419202170068"`

   Note: In this example, the Processing Flag in FID V is set to 1. The terminal must ensure this value is echoed as received.

   The host sends the read mail response to the terminal.

3. Since FID W was included in the previous mail response, the terminal operator sends a read mail request to read any more undelivered mail. In the request, FID V is formatted as follows:

   ".V00419202170068"

4. The host processes the request and determines that no more undelivered mail exists for the terminal. The host formats a read mail response with a response code of 880 (OK, no more data) and does not include FID W. This indicates that all mail has been read. FID V is formatted as follows:

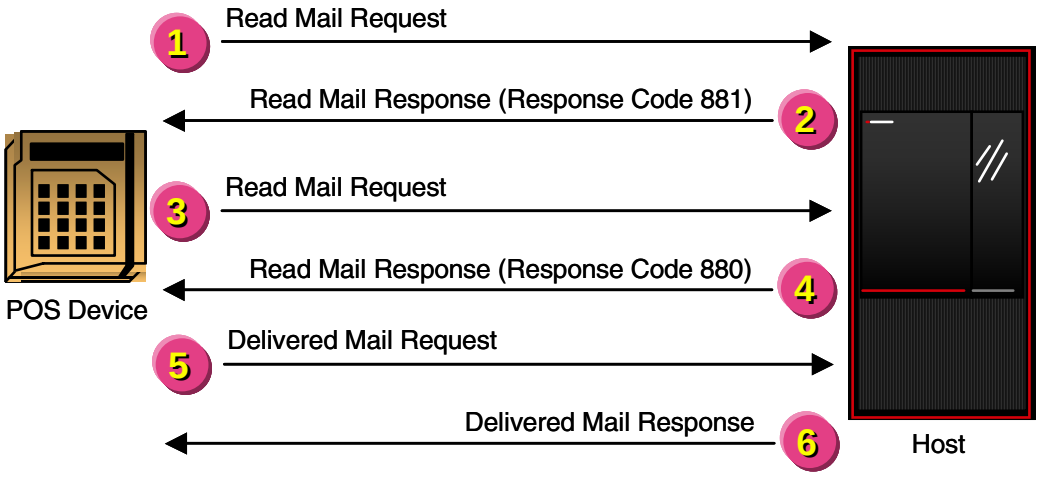
   ".V00419202170068"

   The host sends the read mail response to the terminal.

   Note: The terminal operator can continue to send read next requests to the host as long as the host responses contain a response code of 880 and information in FID W. The absence of FID W indicates that all mail has been read.

Read Mail Request—Multiple Responses

The following diagram illustrates a scenario where there is mail waiting to be delivered to the terminal and the terminal initiates a request to receive the mail. In this example, the text of the mail message is too large for one response.



1. The terminal formats a read mail request and sends it to the host. In this scenario, the terminal requests the first of any undelivered mail using the Mail/Download Key field (FID V). FID V is formatted as follows:

   "V00300000000000"

2. The host retrieves the first piece of undelivered mail and formats a read mail response message. The entire mail message text does not fit within the allotted space of the Mail Text field (FID W) in the response message. Therefore, the response code is set to a value of 881 (mail message received successfully and there is more data for this mail message). FID V is formatted as follows:

   ```
   ".V00419202170068"
   ```

   The host saves the message context for subsequent retrieval and sends the read mail response to the terminal.

3. Since response code 881 was included in the previous mail response, the terminal operator sends a read mail request to the host to read the rest of the mail message. In this request, FID V is echoed from the previous response.

4. The host processes the request, retrieves the next data packet for the mail message, and places the text in FID W. If no more data exists for the mail message, the response code is set to a value of 880. Otherwise, the response code is set to a value of 881.

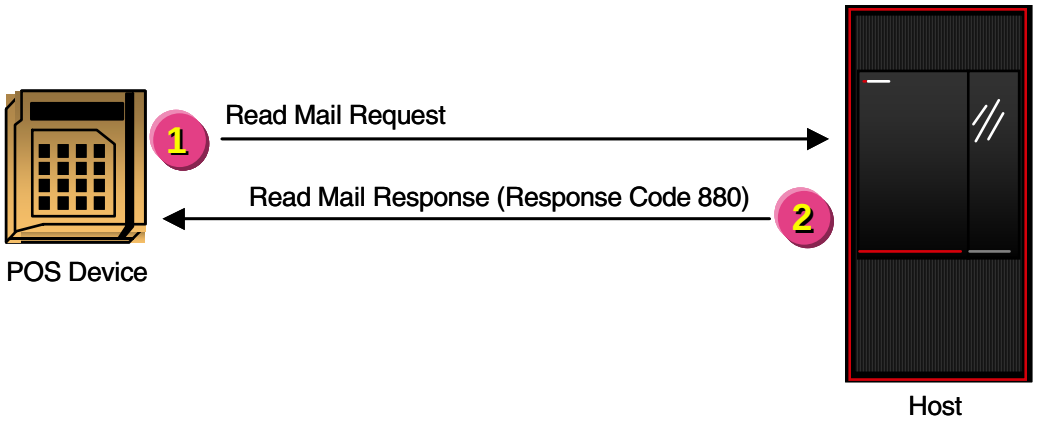
Note: Steps 3 and 4 repeat until no more packets exist for the mail message. Along with the last packet, the host responds with a response code of 880.

5. The terminal sends a mail delivered request to the host to mark the mail as having been delivered. FID V is echoed from the last read mail response.

6. The host marks the mail as delivered for the terminal in the host database and returns a mail delivered response to the terminal.

Read Mail Request—No Mail Stored

The following diagram illustrates a scenario where the terminal initiates a request to receive mail. In this example, no mail is stored for the specified terminal.



1. The terminal formats a read mail request and sends it to the host. In this scenario, the terminal requests the first of any undelivered mail using the Mail/Download Key field (FID V). FID V is formatted as follows:

   `"V00300000000000"`

2. The host processes the request and determines that no more undelivered mail exists for the terminal. The host formats a read mail response with a response code of 880 (OK, no more data) and does not include the Mail Text field (FID W). The host sends the read mail response to the terminal.

Interac Online Payment Transaction Identification Requirements

The following are transaction identification requirements for Interac Online Payment (IOP) transaction support. These transactions are used by institutions which support this type of Internet Payment transaction through the Interac network. These transactions were formerly known as iDebit transactions.

Interac Online Payment transactions are supported through the host with the following fields in the inbound device messages:

- FID D (Application Account Type) must be set to a value of 5.
- FID e (POS Condition Code) must be set to a value of 01.
- FID q (Track 2) must begin with an M to indicate that Track 2 was manually entered.

ACI Worldwide, Inc.

5: Exception Transaction Message Flows

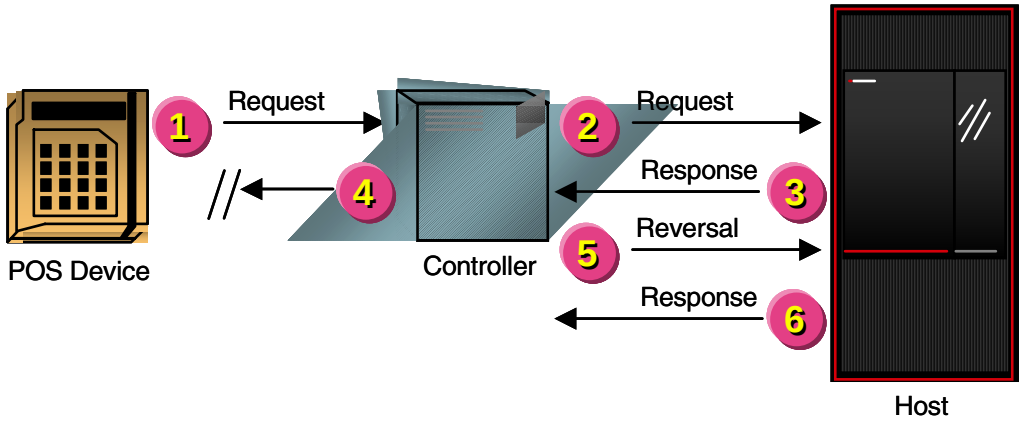
This section contains examples of message flows between a POS device and the host for exception scenarios. Each example includes a diagram illustrating the message flow, followed by a description of each step.

- Terminal- and controller-generated reversals
- Approved Transaction Reversal
- MAC reversal
- Customer-cancellation reversal
- Transaction timeout reversals

Controller Reversal

The diagram below illustrates the transaction message flow in a scenario containing a transaction reversed from a controller. In this transaction, the lines of communication between the terminal and the controller go down, and the controller detects this before it receives a response from the host. The message does not time out. See the "Timeout Reversals" topic later in this section for transaction flows illustrating how reversals are processed when the transaction times out.

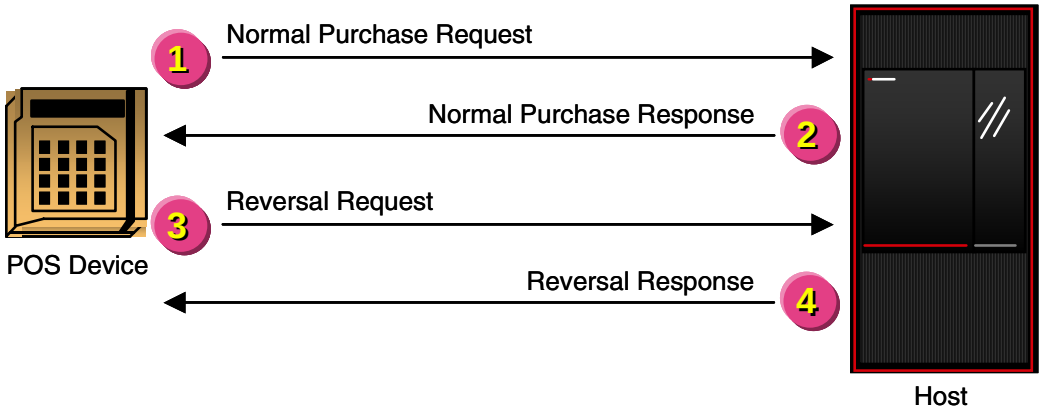
Refer to controller documentation for more information about the communication between the controller and the terminal.



1. The terminal sends an online normal purchase request to the controller.
2. The controller passes the request to the host.
3. The host approves the request, updates terminal totals, and returns an online normal purchase transaction response to the controller.
4. The controller attempts to send the response to the terminal, but the line is down.
5. The controller generates a reversal with a message subtype of C and sends it to the host.
6. The host updates terminal totals and checks the reversal response flag for the terminal in the host database. The flag is set to yes, so the host echoes the reversal request back to the controller. In effect, this echoed message is an acknowledgement to the controller that the host received the reversal request (If the flag had been set to no, the host would not echo the reversal request back to the controller.)

Approved Transaction Reversal

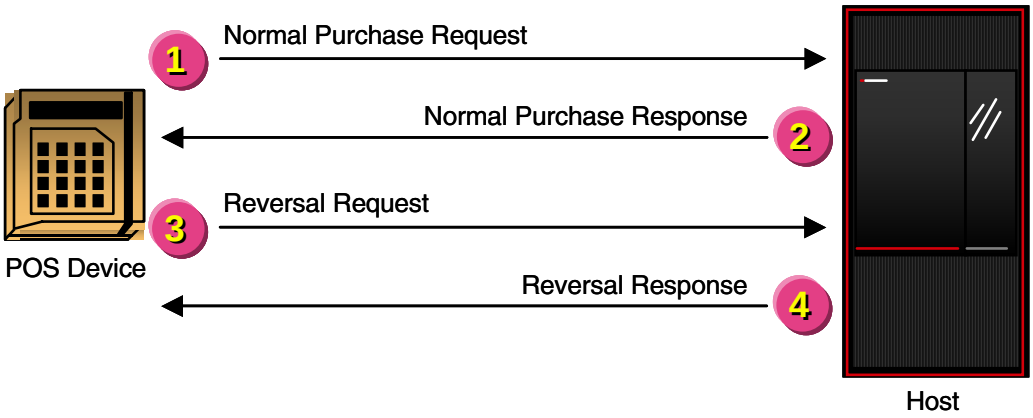
The diagram below illustrates the transaction message flow in the scenario of a reversal of an approved purchase. In this case, the device receives a transaction approval correctly, but for one reason or another the merchant—or the card (in the case of an EMV transaction)—reverses the transaction.



1. The terminal sends an online normal purchase request to the host.
2. The host approves the request, updates terminal totals, and returns an online normal purchase transaction response to the terminal.
3. The terminal receives the approval; however, the merchant—or the card (if this is an EMV transaction)—does not complete the transaction as planned and initiates a reversal. The terminal generates a reversal with a message subtype of C and sends it to the host.
4. The host updates terminal totals and checks the reversal response flag for the terminal in the host database. The flag is set to yes, so the host echoes the reversal request back to the controller. (If the flag had been set to no, the host would not echo the reversal request back to the controller.)

MAC Reversal

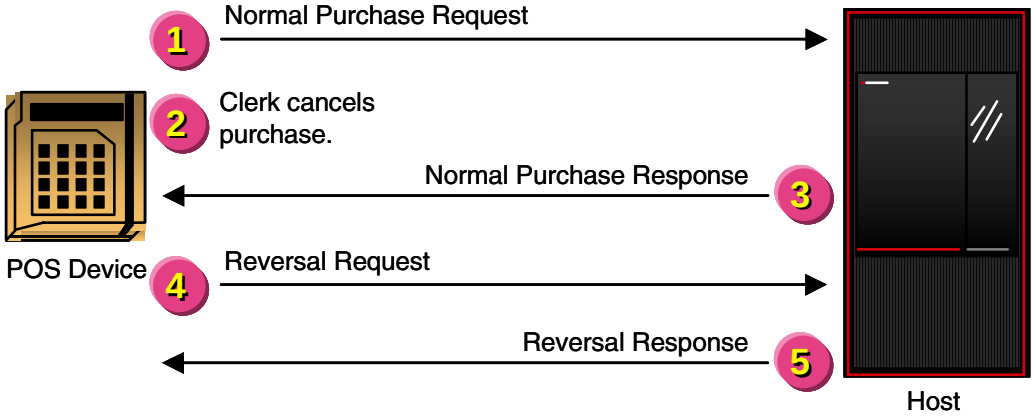
The diagram below illustrates the transaction message flow in a scenario containing a reversal of a message with an invalid message authentication code (MAC).



1. The terminal sends an online normal purchase request to the host.
2. The host approves the request, updates terminal totals, and returns an online normal purchase transaction response to the terminal.
3. The terminal detects a corrupted message while authenticating the message, generates a reversal with a message subtype of R, and sends the reversal to the host.
4. The host updates terminal totals and checks the reversal response flag for the terminal in the host database. The flag is set to yes, so the host echoes the reversal request back to the controller. (If the flag had been set to no, the host would not echo the reversal request back to the controller.)

Customer-Cancellation Reversal

The diagram below illustrates the transaction message flow in a scenario containing a normal purchase transaction that is cancelled by the clerk at the terminal before the host sends a response.



1. The terminal sends an online normal purchase request to the host.
2. The clerk cancels the transaction at the terminal. The terminal waits to send the reversal to the host until after it receives the response to the original transaction.
3. The host approves the normal purchase request, updates terminal totals, and returns an online normal purchase transaction response to the terminal.
4. The terminal generates a reversal with a message subtype of U, and sends the reversal to the host.
5. The host updates terminal totals and checks the reversal response flag for the terminal in the host database. The flag is set to yes, so the host echoes the reversal request back to the controller. (If the flag had been set to no, the host would not echo the reversal request back to the controller.)

Timeout Reversals

This subsection illustrates the following transaction flows in scenarios that contain a reversal at the terminal, controller, or host.

- Timeout of an online transaction at the controller
- Timeout of a store-and-forward transaction at the controller
- Timeout of an online transaction at the host
- Communication failure during a request to the host
- Communication failure during a response to the controller (online transaction); host is aware of the failure
- Communication failure during a response to the controller (store-and-forward transaction); host is aware of the failure
- Communication failure during a response to the controller (online transaction); host is not aware of the failure
- Communication failure during a response to the controller (store-and-forward transaction); host is not aware of the failure
- Timeout of a timeout reversal at the controller

Timeout Reversal Message

When a timeout causes a transaction to be reversed, the controller or terminal (if the terminal is capable of generating a reversal) sends a timeout reversal message to the host. This subsection describes how the host responds to this message.

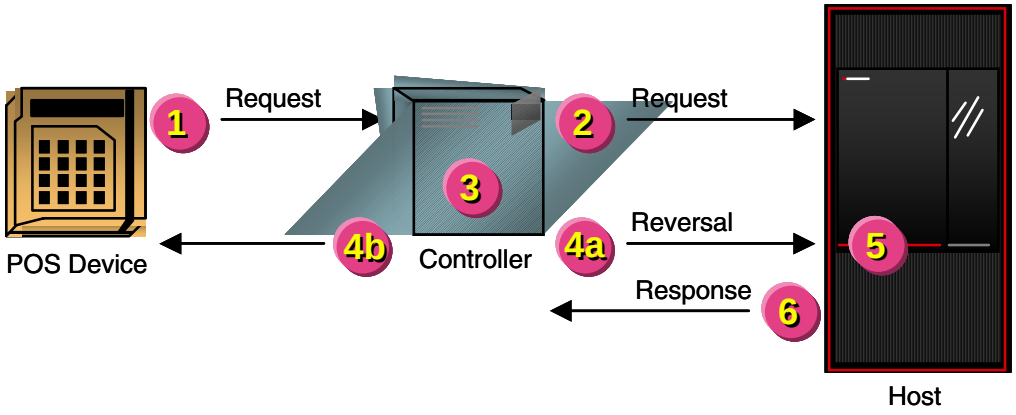
A timeout reversal message contains a standard message header message subtype of T (timeout reversal—online) or A (timeout reversal—advice). The different message subtypes are for the benefit of the controller. For each physical terminal attached to the controller, some controller software uses one logical terminal to process online transactions and one logical terminal to process store-and-forward transactions. The different message subtypes allow the controller to distinguish between online transactions and store-and-forward transactions, so it can route reversal responses to the correct logical terminal. The host processes reversals with message subtypes T and A identically.

The message header of the timeout reversal message must also contain a transmission number. The host uses the transmission number to match the timeout reversal to the transaction it is intended to reverse.

The host responds to a timeout reversal message with the optional timeout reversal response message. The reversal response message is an echo of the reversal request. If your terminal software or controller software supports timeout reversals, the host must be configured to support reversal responses.

Timeout of an Online Transaction at the Controller

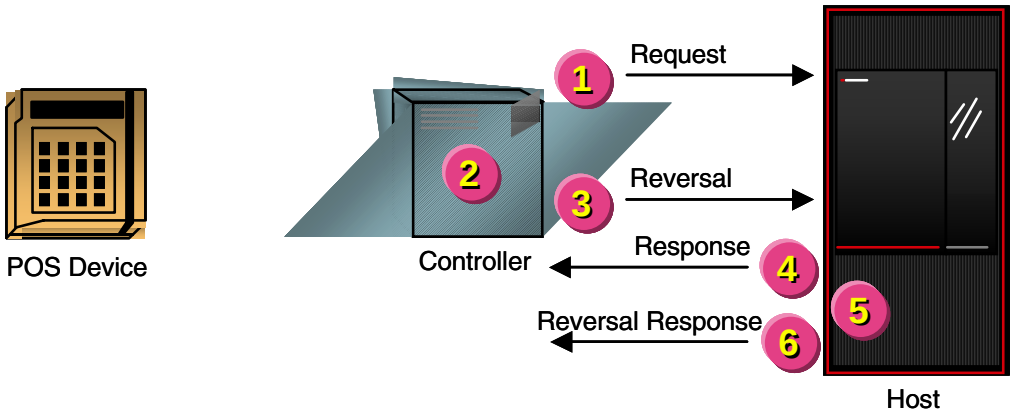
The diagram below illustrates the transaction message flow in a scenario that contains an online transaction that times out at the controller. This transaction begins at the terminal.



1. The terminal sends an online request to the controller.
2. The controller sets a transaction timer and passes the request to the host.
3. The transaction times out at the controller before the response is returned from the host.
4. The controller performs the following:
   a. Generates a timeout reversal message with a message subtype of T and sends the reversal to the host. The transmission number of the timeout reversal message must match that of the online request in step 2.
   b. Responds to the terminal. This leg of processing is now complete.
5. The host reverses the online transaction.
6. The host echoes the reversal request back to the controller.

Timeout of a Store-and-Forward Transaction at the Controller

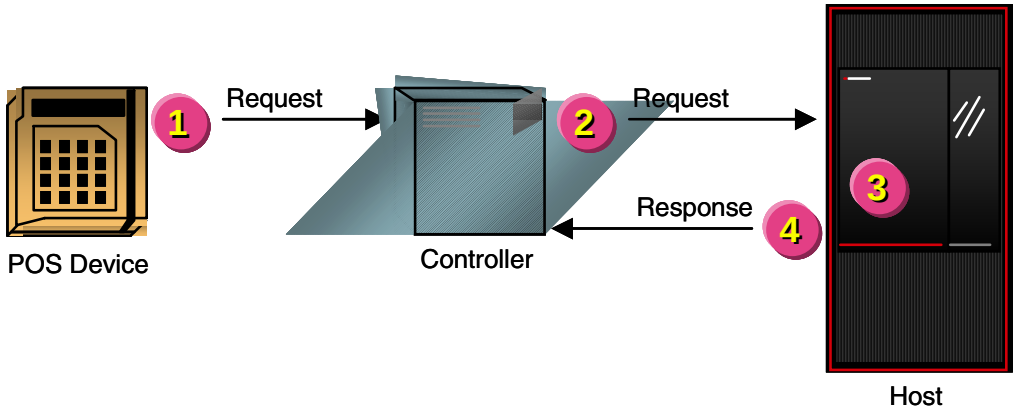
The diagram below illustrates the transaction message flow in a scenario containing a store-and-forward transaction that times out at the controller. This transaction begins at the controller.



1. The controller sets a transaction timer and sends a store-and-forward request to the host with a message subtype of S.
2. The store-and-forward transaction times out at the controller before the response is returned from the host.
3. The controller generates a timeout reversal message with a message subtype of A and sends the reversal to the host. The transmission number of the timeout reversal message must match that of the store-and-forward request in step 1.
4. The host sends a late store-and-forward response message to the controller. The controller drops the message.
5. The host receives the reversal message and reverses the store-and-forward transaction.
6. The host echoes the reversal request back to the controller.

Timeout of an Online Transaction at the Host

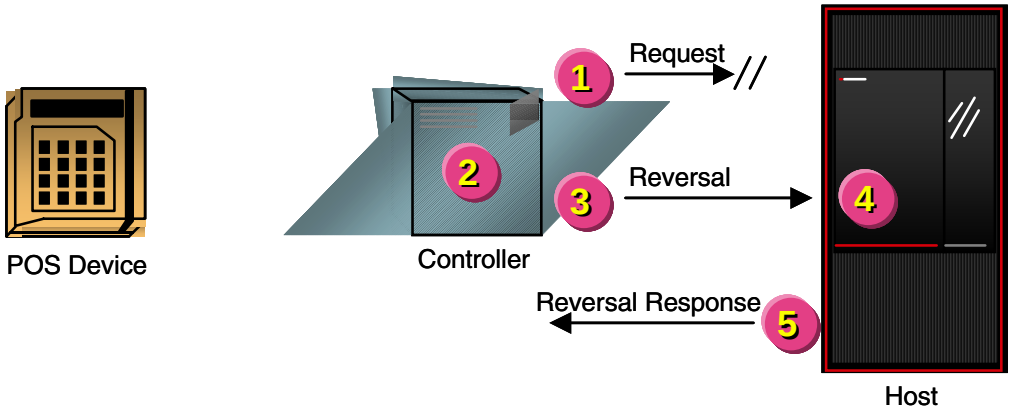
The diagram below illustrates the transaction message flow in a scenario that contains an online transaction that times out at the host. This transaction begins at the terminal.



1. The terminal sends an online request to the controller.
2. The controller passes the request to the host.
3. The transaction times out at the host before an online response can be returned to the controller.
4. The host sends a response with a response code of 810 (timeout) to the controller.

Communication Failure During a Request to the Host

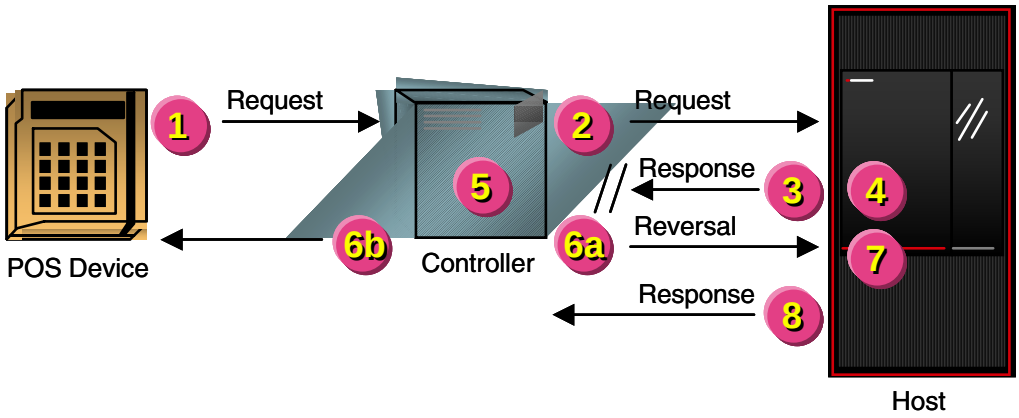
The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a request to the host. The transaction message flow is the same for online and store-and-forward transactions. This transaction begins at the controller.



1. The controller sets a transaction timer and sends a request to the host, but the request fails to reach its destination.
2. The request times out at the controller.
3. The controller sends a timeout reversal with a message subtype of T or A to the host. The transmission number of the timeout reversal matches that of the request sent in step 1.
4. The host determines that the transmission number of the timeout reversal does not match the transmission number of the last transaction processed and drops the timeout reversal.
5. The host echoes a timeout reversal response to the controller.

Communication Failure During a Response to the Controller (Online); Host Aware of Failure

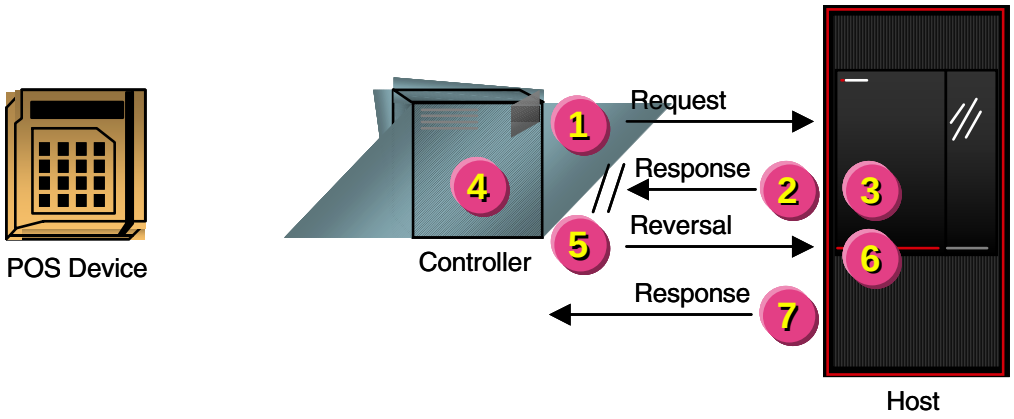
The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for an online transaction. The host is aware of the failure. This transaction begins at the terminal.



1. The terminal sends an online request to the controller.
2. The controller starts a transaction timer and passes the request to the host.
3. The host authorizes the request and sends the online response to the controller, but the response fails to reach its destination.
4. The host detects that the response message failed to reach its destination and reverses the transaction.
5. The transaction times out at the controller.
6. The controller performs as follows:
   a. Sends a timeout reversal message with a message subtype of T to the host. The transmission number of the timeout reversal message must match that of the online request in step 2.
   b. Responds to the terminal. This leg of processing is now complete.
7. The host receives the reversal message from the controller, determines that the transaction has already been reversed, and drops the timeout reversal.
8. The host echoes the timeout reversal response to the controller.

Communication Failure During a Response to the Controller (Store-and-Forward Transaction); Host Aware of Failure

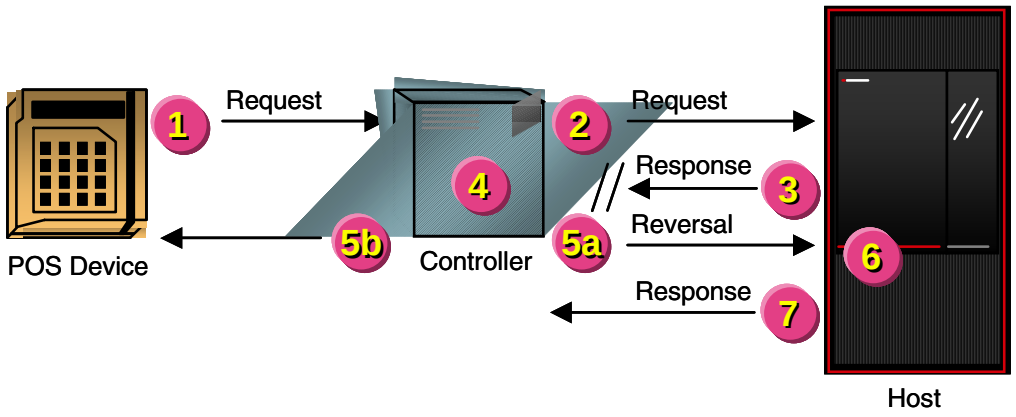
The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for a store-and-forward transaction. The host is aware of the failure.



1. The controller sends a store-and-forward transaction request to the host.
2. The host authorizes the request and sends a store-and-forward transaction response to the controller, but the response fails to reach its destination.
3. The host detects that the response message failed to reach its destination and reverses the transaction.
4. The transaction times out at the controller.
5. The controller sends a timeout reversal with a message subtype of A to the host. The transmission number of the timeout reversal must match that of the store-and-forward transaction request sent in step 1.
6. The host receives the timeout reversal message from the controller, determines that the transaction has already been reversed, and drops the timeout reversal.
7. The host echoes the timeout reversal response to the controller.

Communication Failure During a Response to the Controller (Online); Host Not Aware of Failure

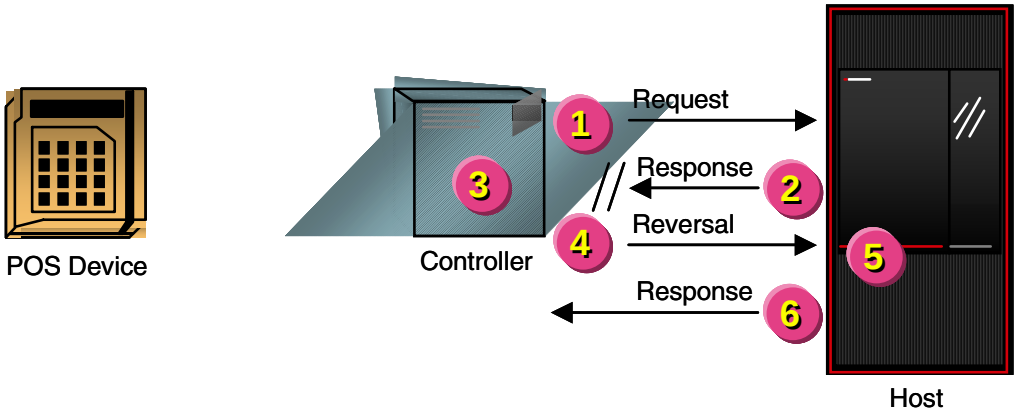
The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for an online transaction. The host is not aware of the failure. This transaction begins at the terminal.



1. The terminal sends an online request to the controller.
2. The controller starts a transaction timer and passes the request to the host.
3. The host authorizes the request and sends the online response to the controller, but the response fails to reach its destination.
4. The transaction times out at the controller.
5. The controller performs as follows:
   a. Sends a timeout reversal message with a message subtype of T to the host. The transmission number of the timeout reversal message must match that of the online request in step 2.
   b. Responds to the terminal. This leg of processing is now complete.
6. The host reverses the transaction.
7. The host echoes the timeout reversal response to the controller.

Communication Failure During a Response to the Controller (Store-and-Forward Transaction); Host Not Aware of Failure

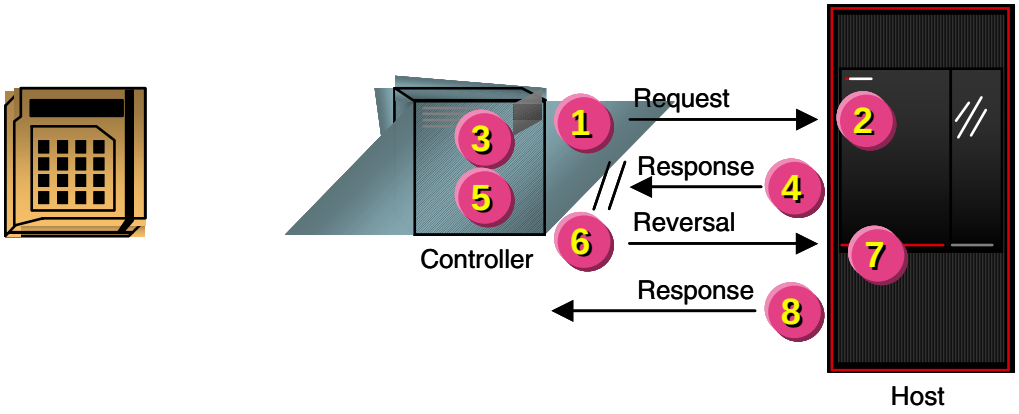
The diagram below illustrates the transaction message flow in a scenario that contains a communication failure during a response to the controller for a store-and-forward transaction. The host is not aware of the failure. This transaction begins at the controller.



1. The controller sends a store-and-forward transaction request to the host.
2. The host authorizes the request and sends a store-and-forward transaction response to the controller, but the response fails to reach its destination.
3. The transaction times out at the controller.
4. The controller sends a timeout reversal with a message subtype of A to the host. The transmission number of the timeout reversal must match that of the store-and-forward transaction request sent in step 1.
5. The host reverses the transaction.
6. The host echoes the timeout reversal response to the controller.

Timeout of a Timeout Reversal Message at the Controller

The diagram below illustrates the transaction message flow in a scenario that contains a timeout of the timeout reversal message at the controller. This transaction begins at the controller.



1. The controller sends a timeout reversal with a message subtype of T or A to the host.
2. The host reverses the transaction.
3. The timeout reversal times out at the controller. The controller sets a delay timer.
4. The host echoes a late timeout reversal response to the controller.
5. The controller drops the late timeout reversal response. The controller delay timer expires.
6. The controller resends the timeout reversal to the host. Steps 2 through 5 repeat until the host echoes a timeout reversal response before the timeout reversal times out at the controller.
7. The host receives the timeout reversal message, determines that the transaction has already been reversed and drops the timeout reversal.
8. The host echoes a timeout reversal response to the controller.

A: Category Code Examples

The Category Code field is contained in the first two bytes of the Mail/Download Key field (FID V). The Category Code field serves the following purposes:

- Notifies the host of initial download requests.
- Indicates to the terminal if a download field being sent from the host to the terminal cannot be fit into a single message. This occurs when the field exceeds the constraints imposed by the maximum response length allowed by the host. In this case, the download field must be sent in multiple messages in order to complete. The Category Code identifies this condition and notifies the terminal that additional information exists (the download field is incomplete). The additional information must be sent in a subsequent response from the host.
- Indicates to the host whether the terminal requests the additional information if a download field does not fit into a single response.
- Indicates to the terminal that the download is complete.

The Category Code consists of two bytes. The first byte is used in responses and the second byte is used in requests.

Responses

In responses, the Category Code indicates whether the last field in the Mail/Download Text field (FID W) contains all the information from that field.

If the first byte of the Category Code is 1, this indicates the last field downloaded in FID W is incomplete. This means the download field cannot be fit into a single message given the constraints imposed by the maximum response length allowed by the host. In this case, the download field must be sent in multiple messages. A value of 1 in the first byte of the Category Code indicates to the terminal that more data exists for a particular download field.

If the first byte of the Category Code is 0, this indicates to the terminal that no more data exists for the particular download field. In this case, the next field is downloaded.

Requests

In requests, the Category Code indicates whether the terminal is required to receive the remaining information from the last field downloaded in FID W (if applicable) or begin receiving information from the next field to be downloaded.

If the second byte of the Category Code is 1, this indicates that the terminal is requesting to receive the remaining data for the download field identified as incomplete in the response (indicated by 1 in the first byte of the Category Code). This is known as a continuation request. The terminal can continue to receive the balance of the data for the incomplete download field until no more data exists as long as the second byte of the Category Code is 1. When no more data exists, the next field is downloaded.

If the second byte of the Category Code is 0, this indicates that the terminal is requesting to begin receiving information from the next field to be downloaded. This can occur in the following situations:

- If the terminal is not requesting a continuation of a current download field identified in the response as being incomplete
- If an incomplete download field is now identified in the response as complete
- If there was no incomplete download field identified in the response (i.e., the download field fit into a single message)

Downloading Complete

When the download is complete, the Category Code is reset to 00 by the host.

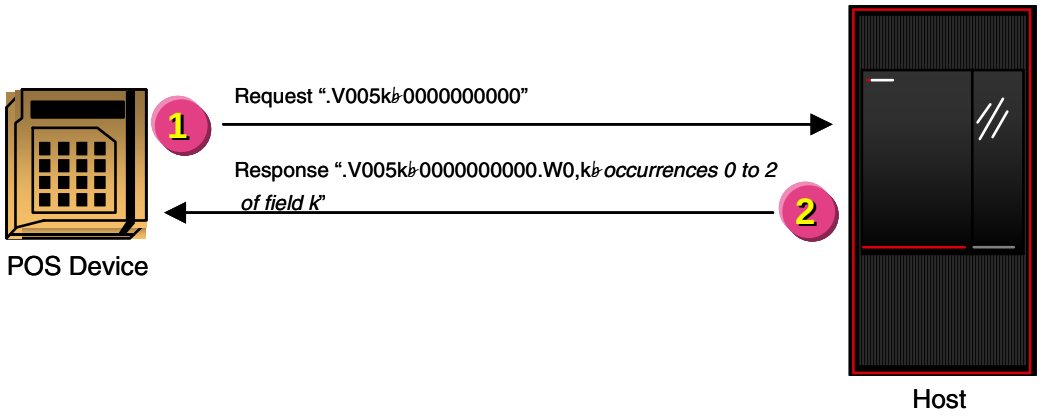
Category Code Examples

The remainder of this appendix provides examples explaining how the Category Code is used in various scenarios. Each example includes a brief summary of the scenario, a graphic illustrating the message exchanges between the host and the terminal during the scenario, and a step-by-step description of each message exchange shown on the graphic.

Data elements are defined and set up by customers in the host database. Each data element that can be downloaded to the terminal is identified with a download field identifier (DID). See section 3 for descriptions of all available DIDs.

Entire Download Field Fits into Response

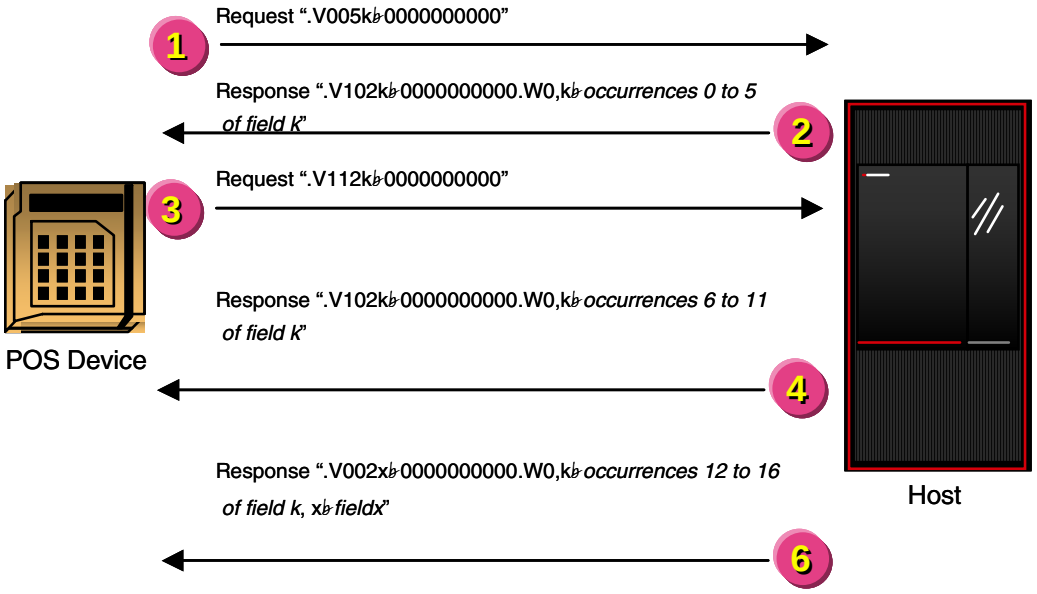
The following example shows a terminal request where the entire download field fits into a single response sent to the terminal from the host. In this scenario, the terminal requests a partial download of the Service download field (DID k), the host responds with the entire contents of the DID k. In this example, there are three services defined in the download database at the host.



1. The terminal requests a partial download of DID k. The Category Code is 00 since this is an initial request. For example purposes, it is assumed that the host has already notified the terminal that a download is in order by sending the terminal a value of 99 in the Category Code field.
2. The host responds to the request by sending as many occurrences of DID k as will fit in a single response message. In this example, the Category Code remains as 00 in the response because the entire contents of DID k is able to be sent in a single response message. The response code in the standard message header is 880 (download has been received in its entirety).

Entire Download Field Does Not Fit into Response (Full Download)

The following example shows a terminal request where the entire download field does not fit into a single response during a full download. In this case, the download field is sent across multiple message responses. In this example, there are 17 services defined in the download database at the host.

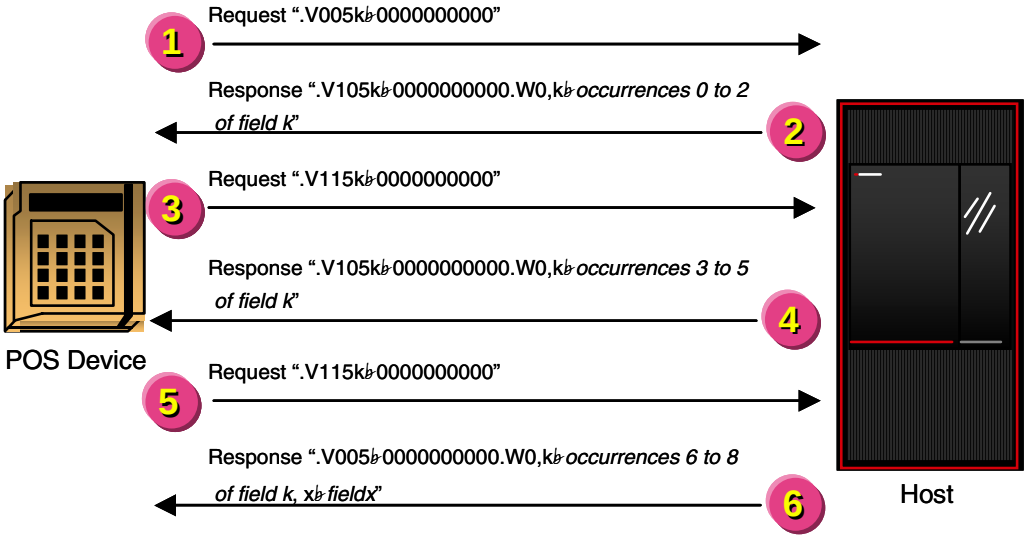


1. The terminal requests a download with no optional data fields. At this point, the Category Code is 00.
2. The host formats a message with six occurrences of the Service download field (DID k) in the response given the message response length constraints of the host. In the example, DID k does not completely fit into the response, so the host changes the first byte of the Category Code to 1. This indicates to the terminal that more information for DID k exists. At this point, the Category Code is 10.
3. The terminal requests more occurrences of DID k by echoing the Mail/ Download Key field (FID V), changing the second byte of the Category Code to 1, and sending the resulting request to the host. This notifies the host to send more occurrences of DID k to the terminal. At this point, the Category Code is 11. The terminal could also have ignored the balance of DID k and moved to the next DID (DID x in the example) by echoing FID V unaltered.

4. The host sends the next six occurrences of DID k in the response. DID k still does not fit into a single response, so the first byte of the Category Code remains as 1. This indicates to the terminal that more information for DID k exists. The second byte of the Category Code is changed back to 0 by the host. The terminal can then change the second byte to 1 if the terminal is requesting more DID k information. However, at this point, the Category Code is 10.
5. The sequence described in steps 3 and 4 continues until the download of DID k is complete or until the terminal does not request any more data.
6. The host sends the last five occurrences of DID k, plus any additional DIDs that will fit into the response. At this point, the host changes the Category Code to 00.

Entire Download Field Does Not Fit into Response (Partial Download)

During partial downloads, the terminal requests a single download field. The following example shows a terminal request where the entire download field does not fit into a single response during a partial download. In this example, the download field requested exceeds the maximum response message length imposed by the host and, therefore, is sent in multiple message responses. In this example, there are nine services defined in the download database at the host.



1. The terminal requests a download for the Service download field (DID k). At this point, the Category Code is 00.
2. The host formats a message with three occurrences of DID k, given the message response length constraints of the host. In the example, DID k does not completely fit into the response, so the host changes the first byte of the Category Code to 1. This indicates to the terminal that more information for DID k exists. The response code in the standard message header is 881 (download received successfully and there is more data for this download). At this point, the Category Code is 10.
3. The terminal requests more occurrences of DID k by formatting a new request, changing the second byte of the Category Code to 1, and sending the resulting request to the host. This notifies the host to send more occurrences of DID k to the terminal. At this point, the Category Code is 11. The terminal could also have ignored the balance of DID k by echoing FID V unaltered.

4. The host sends three more occurrences of DID k in the response. DID k still does not fit into a single response, so the first byte of the Category Code remains as 1. This indicates to the terminal that more information for DID k exists. The second byte of the Category Code is changed back to 0 by the host. The terminal can then change the second byte to 1 if the terminal is requesting more DID k information. However, at this point, the Category Code is 10.
5. The terminal requests more occurrences of DID k by formatting a third new request, changing the second byte of the Category Code to 1, and sending the resulting request to the host.
6. The host sends the last occurrence of DID k. The Category Code is changed to 00 indicating that DID k has completed in the response. The response code in the standard message header is 880 (download has been received in its entirety).

ACI Worldwide, Inc.

B: American Express Standard Industry Formats

The AMEX Data Collection option allows customers to perform draft capture on transactions originating from American Express cardholders. American Express has defined categories under which transactions are placed. These categories are as follows:

- Auto rental
- Lodging
- Restaurant
- General retail
- Oil

For each of these categories, American Express requires different data to be sent from the device. Therefore, American Express has set forth standard industry formats for each category to ensure the data they require is captured by the device and sent to the host. American Express supports a Descriptive Billing File (DBF) that is used to submit charges to American Express without submitting paper. The format of the DBF must conform to the American Express standard. The host is able to recognize these standard industry formats and, subsequently, capture and process transactions sent using them. In conjunction with this field, the customer must set up Standard Industrial Classification (SIC) Codes or Merchant Category Codes for the terminal in the host database. SIC and Merchant Category Codes identify the merchant's line of business.

These formats are for the AMEX Data Collection field (FID 0). The data elements for the industry formats described in this appendix must be included in FID 0 in the order shown.

Valid standard industry formats for each transaction category are shown on the following pages.

Auto Rental

The following format is used for the automobile rental industry.

Field	Length	Description
frmt-cde	9(2)	The American Express Industry Format Code identifies which standard industry format should be used for a particular transaction. This field contains a value of 05 for auto rental transactions.
audit-adj-amt	9(18)	The amount of charges added to or subtracted from the contract after the vehicle was returned (e.g., mileage, damages, discounts, etc.). This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.
agreement-number	X(9)	The invoice number of the rental agreement as issued by the rental agency and signed by the customer.
reference-number	9(8)	A reference number used by American Express to identify the transaction and obtain supporting data from the service establishment for a charge. Valid values are 000000000 through 999999999.
rental-city	X(18)	The city where the rental originated. This field is user-defined.
rental-st	X(2)	The valid two-character alphabetic postal code for the state or province where the rental originated.
rental-dat	9(6)	The date (YYMMDD) that the rental occurred.
rental-tim	9(4)	The time (hhmm) that the rental occurred.
return-city	X(18)	The city where the auto was returned. This value is user-defined.

Field	Length	Description
return-st	X(2)	The state where the auto was returned. This field contains the valid two-character alphabetic postal code for the state or province.
return-dat	9(6)	The date (YYMMDD) on which the auto was returned.
return-tim	9(4)	The time (hhmm) at which the auto was returned.
renter-name	X(20)	The name of the person renting the auto. This value is user-defined.

Lodging

The following format is used for the lodging industry.

Field	Length	Description
frmt-cde	9(2)	The American Express Industry Format Code identifies which standard industry format should be used for a particular transaction. This field contains a value of 11 for lodging transactions.
reference-cde	9(9)	A reference number used by American Express to identify the transaction and obtain supporting data from the service establishment for a charge. Valid values are 000000000 through 999999999.
charge-type	9(1)	A code that indicates the type of charge made. Valid values are as follows: 1 = Lodging 2 = Restaurant 3 = Gift shop 0, 4–9 = Reserved

Field	Length	Description
tab-roc-id	X(10)	The original record of charge (ROC), invoice, or another number used to identify the original transaction. This is an internal American Express code.
tax-amt	9(18)	The amount of sales tax charged for the transaction. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.
arrival-dat	9(6)	The scheduled arrival date (MMDDYY).
depart-dat	9(6)	The scheduled departure date (MMDDYY).
room-rate	9(18)	The per diem rate charged for the customer's stay at the establishment. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.
program-cde	9(1)	A code indicating the reason for the charge. If there are no special circumstances that require a code, a value of 1 is placed in this field. Valid values are as follows: 1 = Used if not other code pertains 2 = Assured reservation—no show 3 = CARDeposit 4 = Delayed charge 5 = Express service 6 = Assured reservation 0, 7–9 = Reserved

Restaurant

The following format is used by the restaurant industry.

Field	Length	Description
frmt-cde	9(2)	The American Express Industry Format Code identifies which standard industry format should be used for a particular transaction. This field contains a value of 12 for restaurant transactions.
reference-cde	9(9)	A reference number used by American Express to identify the transaction and obtain supporting data from the service establishment for a charge. Valid values are 000000000 through 999999999.
charge-cde1	X(4)	A code mapped by the host to a description that identifies the transaction associated with the charge. This value is defined by American Express.
tab-roc-id	X(10)	The original record of charge (ROC), invoice, or another number used to identify the original transaction. This is an internal American Express code.
description-cde	X(1)	A code that describes the nature of the transaction. Valid values are as follows: 0 = Food 1 = Food and beverage 2 = Gift certificate 3–9 = User-defined A–Z = User-defined
tax-amt	9(18)	The amount of sales tax charged for the transaction. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.

Field	Length	Description
food-amt	9(18)	The total cost of food, or food and beverage if these charges are combined on the original record of charge (ROC). This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.
beverage-amt	9(18)	The total cost of beverages, if itemized. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.
tip1-cde	X(1)	A code mapped by the host to a description that designates an employee receiving a tip.
tip2-cde	X(1)	A code mapped by the host to a description that designates a second tip itemized on the original record of charge (ROC).
tip1-amt	9(18)	The total amount of the tip described in the tip1-cde field. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.
tip2-amt	9(18)	The total amount of the second tip described in the tip2-cde field of the American Express token. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.

General Retail

The following format is used by the general retail industry.

Field	Length	Description
frmt-cde	9(2)	The American Express Industry Format Code identifies which standard industry format should be used for a particular transaction. This field contains a value of 20 for general retail transactions.
tax-amt	9(18)	The amount of sales tax charged for the transaction. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.
reference-cde	9(9)	A reference number used by American Express to identify the transaction and obtain supporting data from the service establishment for a charge. Valid values are 000000000 through 999999999.
charge-cde1	X(4)	A code mapped by the host to a description that identifies the transaction associated with the charge. This value is defined by American Express.
tab-roc-id	X(10)	The original record of charge (ROC), invoice, or another number used to identify the original transaction. This is an internal American Express code.
charge-cde2	X(4)	A code mapped by the host to a description that identifies the transaction associated with the charge. This value is defined by American Express.
charge-cde3	X(4)	A code mapped by the host to a description that identifies the transaction associated with the charge. This value is defined by American Express.

Field	Length	Description
charge-cde4	X(4)	A code mapped by the host to a description that identifies the transaction associated with the charge. This value is defined by American Express.
charge-cde5	X(4)	A code mapped by the host to a description that identifies the transaction associated with the charge. This value is defined by American Express.

Oil

The following format is used by the oil industry.

Field	Length	Value
frmt-cde	9(2)	The American Express Industry Format Code identifies which standard industry format should be used for a particular transaction. This field contains a value of 21 for oil company transactions
reference-cde	9(12)	A reference number used by American Express to identify the transaction and obtain supporting data from the service establishment for a charge. Valid values are 000000000000 through 999999999999.
charge-cde1	X(4)	A code mapped by the host to a description that identifies the transaction associated with the charge. This value is defined by American Express.
tab-roc-id	X(10)	The original record of charge (ROC), invoice, or another number used to identify the original transaction. This is an internal American Express code.

Field	Length	Value
tax-amt	9(18)	Indicates the amount of sales tax charged for the transaction. This field has an implied decimal two positions from the rightmost character. If not used, this field contains 0.

ACI Worldwide, Inc.

Index

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

X

ACI Worldwide, Inc.