vantiv



Interface Specification

March 2016

CONFIDENTIAL

If you, as the reader of this document, are not the intended recipient, please be advised that any use, dissemination or copying of the concepts or contents of this document is respectfully prohibited. If you received this document in error please notify the sender immediately; thank you.

Document History

Date	Name	Change Description	Version
03.03.2016	Lisa Killigrew	rebranded document to Vantiv Integrated Payments	2.7.7
2.26.2016	Marc Jorgensen	Updated dates and versions for public release	2.7.7
12.01.2015	Manish Anavkar	 Added TokenID as an optional input parameter to CreditCardReversal using TransArmor token 	2.7.7
07.29.2015	Manish Anavkar Jeff Gross	 Added TAProviderID to Token class. Updated CreditCardAuthorization and CreditCardSale request and response for Token object. Added TokenProvider enumeration table. Added EncryptedFormat of Format6 and Format7. Added TokenCreate and TokenCreateWithTransID methods 	2.7.6
07.01.2015	Lisa Killigrew	Revised document content and format for improved usability.	2.7.5
02.19.2015	Kyle Ferrio Jeff Gross	 Added EnhancedBINQuery method Added LaneNumber field to descriptions of <u>Terminal</u> class and applicable methods Removed MagneprintDataDecrypt method 	2.7.5
11.06.2014	Jeff Gross	 Added EncryptedFormat of Format5. Added Check/ACH Null Simulator tests. 	2.7.4
04.23.2014	Jeff Gross	 Updated AVS Response Code A and Y descriptions. Updated description of CreditCardAdjustment method. Added ConvenienceFeeAmount to TransactionSetup method. Added Amex Level III input fields to EnhancedData ExtendedParameters. Removed TotalAuthorizedAmount from CreditCardReversal method. Removed references to documentation no longer available. 	2.7.3
02.24.2014	Jeff Gross	 Added ConvenienceFeeAmount to Transaction class. Added ConvenienceFeeAmount to CreditCardSale, CreditCardAuthorization, CreditCardAuthorizationCompletion, CreditCardForce, CreditCardReturn, and CreditCardCredit methods. Added DebitCardPinlessSale and DebitCardPinlessReturn methods. 	
12.11.2013	Jeff Gross	 Updated LodgingSpecialProgramCode Sale enumeration description. Removed TransactionStatus as an optional TransactionQuery input parameter Added TransactionStatusCode as an optional TransactionQuery input parameter Clarified Healthcare class to state that the Healthcare class is 	2.7.1

If you, as the reader of this document, are not the intended recipient, please be advised that any use, dissemination or copying of the concepts or contents of this document is respectfully prohibited. If you received this document in error please notify the sender immediately; thank you.

Date	Name	Change Description	Version
		used for Visa and MasterCard FSA cards only. • Updated TerminalEncryptionFormat description	
5.6.2013	 Updated document references from Payment Account Secure Storage to TransForm Tokenization Updated TerminalSerialNumber to Conditional and updated description 		2.7.0
1.17.2013	Jeff Gross	 Updated Healthcare ExtendedParameters descriptions Updated ItemDiscountRate description 	2.6.9

Table of Contents

How to Use this Document	7
Product Overview	8
What is Express?	8
Business Needs Handled by Express	8
Services Offered	
Technical Architecture	
Application Program Interface (API) Overview	11
Transactions Supported	12
Credit	
Debit	
Check	
Business Rules and Process Flow for Transactions	
Hosted Payments	
TransForm® Tokenization (Payment Account Secure Storage)	
Account Updater	
Security Methods and Procedures	
Interface Specification	
Common Fields Descriptions	
EnhancedData Extended Parameter Fields	28
Functionality	33
Duplicate Checking	
Recurring Transactions	
Scheduled Tasks	
Check Processing	
Level III Processing	35
Methods	
Batch	
Credit	
Debit	
Check	
Reporting	
Services	
System Hosted Payments	
BIN Management	
Classes	
Address Class	
Autress Class	137
Application Class	158
AutoRental Class	158
Batch Class	160
BIN Class	161
Card Class	161
Credentials Class	
DemandDepositAccount Class	163
ExtendedParameters Class	164

Healthcare Class	164
dentification Class	
Lodging Class	166
Paging Class	167
Parameters Class	167
PaymentAccount Class	168
PaymentAccountParameters Class	169
Response Class	170
ScheduledTask Class	171
ScheduledTaskParameters Class	
Terminal Class	
Transaction Class	
Token Class	
TransactionSetup Class	
Enumerations	
☐ CardholderPresentCode	
EleckType	
■ CVVResponseType	179
■ DDAAccountType	179
■ ReversalType	
■ BooleanType	
■ CardPresentCode	
CVVPresenceCode	
■ BatchCloseType	
■ BatchIndexCode	
■ BatchGroupingCode	
■ BatchQueryType	
■ MarketCode	
■ MotoECICode	
■PaymentAccountType	
■ RunFrequency	
TerminalCapabilityCode	
FreminalEnvironmentCode	
TerminalType	
TransactionStatusCode	
ExtendedBooleanType	186
ExtendedStatusType	
ExtendedRunFrequency	
ScheduledTaskRunStatus	187
Povice	197

Fransaction Setup Method	187
Healthcare[First][Second][Third][Fourth]AccountType	188
Healthcare[First][Second][Third][Fourth]AmountType	188
Fealthcare[First][Second][Third][Fourth]AmountSign	
ElodgingSpecialProgramCode	
ElodgingChargeType	
■ LodgingPrestigiousPropertyCode	
AutoRentalVehicleClassCode	
AutoRentalDistanceUnit	
AutoRentalDistanceOffit AutoRentalAuditAdjustmentCode	
_	
PASSUpdaterBatchStatus	
PASSUpdaterOption	
	192
■ DeviceInputCode	192
■ TerminalEncryptionFormat	193
■ EncryptedFormat	193
₽ TokenProvider	
Appendix	194
Express Response Codes	
AVS Response Codes	
CVV Response Codes	196
CAVV Response Codes	196
Commercial Card Response Codes	196
Sample Code	197
NULL Processor	
Terminal Settings	200
Interface URL's	200
Element Certification Team	200

How to Use this Document

This document provides a detailed description of how to integrate with the *Vantiv Integrated Payments Express* payment processing platform. This document is intended for:

• IT decision makers who want to know more about the features and capabilities of *Express* (what services are offered, what needs it addresses, how it is structured, etc.) It will also help to clarify what they will need to do to use *Express* in their own custom POS applications. This specification has useful background knowledge for those who are new to the payment processing industry.

The following sections of this document are helpful for IT decision makers:

Product Overview

- What is Express
- Business Needs Handled by Express
- Services Offered
- Technical Architecture
- Application Program Interface (API) Overview

Transactions Supported

- Business Rules and Process Flow for Transactions
- Security Methods and Procedures
- **Developers and experienced technical people** who wish to integrate *Express* into their software applications to handle a variety of payment types. Information is included to help developers design, implement, test, and support their applications.

Experienced technical people will want to review the <u>transaction flow diagram</u> followed by studying the <u>Interface Specification</u>. This is a quick way to help you get familiar with the interface elements.

The sections after that provide complete descriptions of the <u>Methods</u>, <u>Classes</u> and <u>Enumerations</u> available to you when using *Express*. These can aid in quickly developing your payment processing code.

The <u>Appendix</u> contains a variety of information such as response code descriptions, suggested terminal settings, sample code, etc.

We highly recommend you periodically check for the latest version of this document at: http://www.elementps.com/docs/Express-Interface-Specification.pdf

Product Overview

What is Express?

Express is a highly secured, scalable, and reliable payment processing platform providing low cost and high speed transaction processing for card-present (such as retail, auto rental, restaurant, etc.) and card-not-present (ecommerce, mail order, telephone order) merchants.

- It provides credit card, debit card, and check authorizations processing via the internet for all card types including Visa, MasterCard, AMEX, Discover, Diners, and JCB.
- Integrating Express with your software provides merchants end-to-end encryption of cardholder data, the ability to store sensitive data off-site to meet PCI DSS compliance requirements, and comprehensive Web-based reporting on their payment card transactions.
- As an added benefit, because we develop and support the platform entirely in-house, there are no third parties to chase for support.



From a technical standpoint, *Express* is a server-side Web service application where all communication takes place over secure HTTPS links to the internet. Communication is handled via XML or SOAP, which are both simple and effective methods for handling payment processing via the internet. No distribution of source code or DLLs is required, and any device that can originate an HTTPS connection can utilize payment processing through *Express*.

Integrating and certifying POS and/or eCommerce applications to *Express* is quick and easy. The *Express* processing platform doesn't require you or your customers to download, run or support additional code.

Business Needs Handled by Express

Express provides credit card, debit card, and check authorization processing via the internet for all card types including Visa, MasterCard, American Express, Discover, Diners, and JCB. Businesses can integrate this with their software or other customized applications to handle these types of payments in a safe, secure way.

Express supports transactions for the following merchant types:

Retail

- Auto Rental
- eCommerce
- Healthcare
- Direct Marketing
- Lodging

Express has the versatility to handle additional merchant types. If you do not see what you need, or have any questions, please contact: certification@vantiv.com

Recurring transactions can be set up for monthly billing of a client's card, *TransForm* Tokenization can be used for very secure, low risk payment processing, and *Express* can alert you to any transactions that have been accidently sent twice. *Express* also offers the service of **Hosted Payments** which takes merchants out of scope of having to collect and store sensitive customer card information.

Services Offered

Hosted Payment service

Element offers a *Hosted Payments* solution which is specifically designed to remove business applications from the scope of PA-DSS and also limit the financial risk of merchants through the use of fully integrated payment solutions. The business application is responsible for collecting all of the non-sensitive data needed to perform a payment transaction, while Element's *Hosted Payments* solution is responsible for collecting, storing, processing, and transmitting all of the sensitive cardholder data.

TransForm[®] Tokenization

TransForm Tokenization reduces the financial risk associated with storing sensitive cardholder data by eliminating the need for merchants to store the data at all. Merchants can create payment records on Element's server by initially providing the cardholder account number, expiration date, etc. A unique identifier is generated and provided to the business application, which acts as a pointer for all subsequent transactions and can be stored. The vulnerable cardholder information can never be obtained by using the pointer, and the pointer is of no value to anyone attempting to steal sensitive information. This method takes your application a step beyond tokenization in providing data security.

Recurring Transactions

Recurring transactions are a unique type of transaction where consumers authorize merchants or solution providers to bill a specific card on a regular basis (e.g. monthly membership fees). Once a merchant has scheduled a payment within their recurring billing system, the system submits the transaction details and a flag indicating that it is a recurring transaction, along with the previously-acquired pointer, to the Express processing platform. Neither the merchant nor the recurring billing system has access to sensitive cardholder information, thus reducing the financial risk associated with storing this recurring information locally. Note that all transactions are initiated by the merchant's software.

Scheduled Transactions

Scheduled transactions can also be setup on *Express*. These are transactions that occur automatically without manual intervention from the merchant's billing system, and include both recurring payments and one-time future payments.

Duplicate Checking

This function ensures that no single transaction is accidentally submitted and processed twice by the merchant. Duplicate checking is done in real-time as transactions are submitted for authorization. For a merchant account, if *Express* sees the same card number for the same amount on the same day in the same batch as a previously-approved transaction, it will automatically reject that transaction and will not allow it to process a second time. The response will indicate the error.

Check Processing

Express can handle check processing just as easily as credit / debit cards to make your business more versatile.

Technical Architecture

Express is designed as a Service Oriented Architecture (SOA), and provides both a platform and language independent Application Program Interface (API). Virtually any device that can access the internet can integrate to Express. See the section on Integration for more details.

Advantages and Features

- Simple
- Secure
- Solid Design
- Reliable
- Advanced Technology

As a software provider, you've seen it all too often: connecting with another application or an online service provider means an investment in software development kits, training your staff in a new interface and maintaining, troubleshooting and updating the interface over time. Some payment processing solutions even require you, or your merchant, to buy, run and maintain additional software.

We recognized the inherent weaknesses of these approaches and so built our platform to support standard XML and Web services interfaces. This robust and flexible design allows you to quickly and easily integrate your software with the *Express*. Our Service Oriented Architecture means we can easily add new features, such as support for new security requirements, and these new features are immediately available to you.

Integration

Integration is quick and easy. Developers can choose from either an XML based Web Services (SOAP) or a pure XML interface. Both interfaces support the same transaction set (with only a couple of minor exceptions); the choice is left to the discretion of the developer.

▼ Note Extended ASCII characters (for use in foreign names, etc.) are not supported within the transaction request, and if used will cause the request to fail. More details are provided under *System Requirements* below.

System Requirements

- Any system which can initiate an HTTPS request over the internet can communicate with Express to run transactions. This could be a PC with practice management software, a credit card terminal, or any POS application.
- Express is designed as a Service Oriented Architecture (SOA) and provides both a platform and language independent Application Program Interface (API).
- Express requires all transactions to be sent using standard 128-bit Secure Socket Layer (SSL) encryption. Additionally, every transaction must be authenticated. Authentication is achieved through use of an **AccountID** and **AccountToken** issued by Element.
- As a financial service provider, Vantiv Integrated Payments is required to comply with the Payment
 Card Industry (PCI) Data Security Standard (PCI DSS). The PCI Data Security Standard comprises
 twelve requirements pertaining to information security, including strong access controls, configuring
 security parameters appropriately on systems, tracking and monitoring data access, and restricting
 physical access to the systems containing card data. The full details of the PCI requirements can be
 found at www.pcisecuritystandards.org

Application Program Interface (API) Overview

Interfaces Supported

• Simple Object Access Protocol (SOAP)

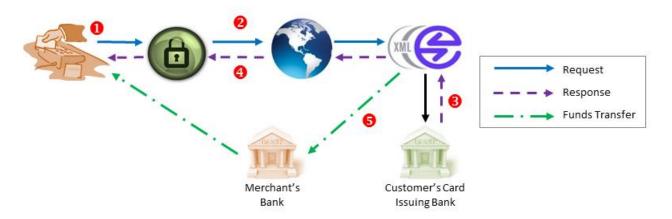
Express supports version 1.1 of the W3C SOAP specification. Using this as your interface, payment methods can be invoked remotely on Element's servers in a manner similar to Remote Procedure Calls.

Extensible Markup Language (XML)

Requests can also be sent to the Express servers via the POST method in the HTTPS protocol to send XML data. XML data will then be returned in the reply.

Transaction Flow

The below provides a high level view of the communication path when a merchant runs a POS payment transaction:



- 1. The merchant swipes the customer's card, which generates an authorization request based on the purchase price.
- 2. The request is sent via secured payment data along the internet to Express.
- 3. Express forwards the authorization request to the customer's card issuing bank. The issuing bank verifies the purchase price against the customer's available credit amount, then generates either an an "approval" or a "decline" response to Express.
- 4. *Express* forwards the response back to the merchant's terminal, and the transaction is either accepted or declined.
- 5. If the transaction is approved, a few days later the funds are transferred to the merchant's bank.

Transactions Supported

Express supports Credit, Debit, and Checking transactions. Each of these are described in detail below.

Credit

Sale

The *Sale* transaction is the most basic and most common transaction, used for the sale/purchase of goods or services. This type of transaction compares the cardholder's credit limit to the amount specified in the transaction. If the amount specified is available, the transaction is approved by the card issuer and an approval code is returned. If the amount is **not** available, or the transaction is denied for any other reason by the issuer, an error message containing the reason is returned.

See *CreditCardSale* for details.

Authorization

An Authorization transaction is used to verify funds when the total amount of the purchase is unknown.

See CreditCardAuthorization for details.

Authorization Completion

An *Authorization Completion* transaction is used to complete the final amount of a previously *Authorized* transaction.

Example: Restaurants use *Authorization Completions* to enter in the final total amount after a customer has added a tip and signed the receipt.

An *Authorization Completion* can be entered for any amount up to an allowable excess limit. The allowable excess limit is determined by the card association.

If the final amount exceeds the *Authorization* allowable excess limit, then merchant can either enter a *Sale* or *Force* transaction for the full amount.

See CreditCardAuthorizationCompletion for details.

AVS Only

The Address Verification Service (AVS) Only transaction may be used by merchants to determine whether to accept a transaction. This type of transaction only returns AVS information and does not request authorization. The format of the AVS information is exactly the same as the format used for Sale transactions.

See <u>CreditCardAVSOnly</u> for details.

Credit / Return

A *Credit* transaction is used to refund a cardholder for a previous transaction. A *Return* transaction is like a *Credit* transaction except it requires a *TransactionID* from the original dependent transaction.

Example: A customer returns merchandise and wants a refund.

See CreditCardCredit and CreditCardReturn for details.

Reversal

There are three types of *Reversal* transactions:

- 1. Communications/System Reversals
- 2. Full Reversals
- 3. Partial Reversals

A **Communications/System Reversal** is generated when there is a problem delivering the response back to the POS. It is the transaction requestor's responsibility to generate a *Reversal* transaction for all transactions that do not receive responses.

A Point of Sale (POS) Generated Reversal or **Full Reversal** may be initiated to remove an approved transaction from a batch, even when the cardholder is not present. Reversals, however, must be done while the batch is open.

A **Partial Reversal** transaction decreases the total authorized amount of a previously authorized transaction. Partial reversals are required by some card associations to bring the authorized and clearing amounts into specific tolerances.

See CreditCardReversal for details.

Force

Generally, *Force* transactions are used to insert previously authorized, but not captured, transactions into a batch. The following examples illustrate when to use a *Force* transaction:

- 1. **Timeout** If the merchant is unable to perform an online transaction.
- 2. **Referral** If the merchant receives an online referral message in response to an online transaction (i.e. Call ND, Call AE, etc).
- 3. **Al Responses** The *Al Response* (Approved with positive ID) usually results from a person using a card for the first time.

See *CreditCardForce* for details.

Void

A *Void* transaction is used to completely back out a previous *Sale* transaction. This applies to all transactions that were entered into the current batch, and only for the full dollar amount of the original transaction.

See CreditCardVoid for details.

Adjustment

The *Adjustment* transaction is used to edit the information on the host for a previously completed credit card transactions (e.g. Level III Line Item Detail).

Depending on the platform, this transaction can also be used to correct the Host batch to match the merchant's paper sales slips when a discrepancy exists.

See CreditCardAdjustment for details.

Incremental Authorization

Incremental Authorization transactions are used to authorize an additional amount to an original **Auto**Rental or Lodging industry authorization transaction. This method is used on the TSYS and FDC platforms only. Only Visa cards support incremental authorizations.

See CreditCardIncrementalAuthorization for details.

Debit

Sale

The *Sale* transaction is the most basic and most common transaction, used for the sale/purchase of goods or services. This type of transaction compares the cardholder's available funds to the amount specified in the transaction. If the amount specified is available, the transaction is approved by the cardholder's bank and an approval code is returned. If the amount is **not** available or the transaction is denied for any other reason by the bank, an error message containing the reason is returned.

See DebitCardSale for details.

Return

A *Return* transaction is used to refund a cardholder for a previous transaction. Debit card *Returns* must be done online. For security reasons, this method may not be available on your merchant account. Contact the Express Certification team for further details.

See *DebitCardReturn* for details.

Reversal

The most common type of *Reversal* available for Debit transactions is a **Communication/System Reversal**. A *Communications/System Reversal* is generated when there is a problem delivering the response back to the POS. It is the transaction requestor's responsibility to generate a *Reversal* transaction for all transactions that do not receive responses.

See DebitCardReversal for details.

Check

Verification

A *Verification* transaction is used to verify consumer information as well as account/check information. This method is only for verification. It does not originate the transaction or move funds.

See CheckVerification for details.

Sale

The Sale transaction is the most basic and most common transaction, used for the sale/purchase of goods or services.

See CheckSale for details.

Credit / Return

A *Credit* transaction is used to refund a consumer for a previous transaction. A *Return* transaction is like a *Credit* transaction except it requires a *TransactionID* from the original dependent transaction.

Example: A customer returns merchandise and wants a refund.

See CheckCredit and CheckReturn for details.

Void

A *Void* transaction is used to completely back out a previous *Sale, Credit, or Return* transaction. This applies to the full dollar amount of the original transaction.

See CheckVoid for details.

Reversal

A **Communications/System Reversal** is generated when there is a problem delivering the response back to the POS. It is the transaction requestor's responsibility to generate a *Reversal* transaction for all transactions that do not receive responses.

See CheckReversal for details.

Business Rules and Process Flow for Transactions

We have designed *Express* to be not only highly secure, but also language and platform independent. This means that a wide variety of payment processing devices, computer and software systems can use *Express*. New applications can easily be developed using any of a number of different programming languages to integrate with *Express*. Below is an example of one way this can work:

- An authorization request is generated from either a POS device, a software program, or a website to
 verify that a credit card has sufficient funds (credit) available to cover the amount of a sale. If the
 card is physically present, such as in a store, it is swiped through the POS device and the card holder
 information is read from the magnetic stripe.
- If the card is not present (mail order, telephone order, internet sale, etc.), the information is manually keyed in, and the CVV code is used for additional verification. The address information should be verified first to determine accuracy and deter fraud. A code is then returned with the authorization result that indicates the level of accuracy of the address match and helps secure the most favorable interchange rates.
- For all types, transaction information is encrypted and sent as secured payment data to through the internet. The authorization request is routed by *Express* to the customer's Card Issuing bank, which verifies that the account has sufficient funds to cover the sale amount and that no fraud or data entry errors have been detected. If the request is accepted, an authorization response of "Approval" is sent back to the POS device or software program and the sale can proceed. If the authorization has been rejected, a "Decline" response is sent and another form of payment may be needed.
- For approvals, funds are reserved in the customer's credit card account equal to the total purchase price. Settlement takes place each night, and within a few days, funds from the sale are transferred to the merchant's bank account.

Hosted Payments

How does it work?

Element offers two types of *Hosted Payments*:

- 1. Hosted Payment Window for distributed software applications
- 2. Hosted Payment Pages for Web-based software applications

Both types of *Hosted Payments* work essentially the same way. The software vendor's application is responsible for collecting all of the non-sensitive data needed to perform a payment transaction, while the *Hosted Payments* application is responsible for collecting, storing, transmitting, and processing all of the sensitive cardholder data.

Hosted Payment Window (for distributed software applications)

The *Hosted Payment Window* is designed for distributed software applications through the use of an embedded browser control. The below outlines the steps to process a *Hosted Payment Window* transaction with a typical POS application:

- 1. The POS collects all of the non-sensitive data related to the purchase.
- 2. The POS submits this non-sensitive data by calling the *TransactionSetup* method. A *TransactionSetupID* (GUID) is returned and used to identify the specific purchase from that point forward.
- 3. The POS displays a window/form that has an embedded browser control. This browser control navigates to Element's *Hosted Payments* system (making sure to include the appropriate *TransactionSetupID*): https://certtransaction.hostedpayments.com/?TransactionSetupID=[InsertHere].
- 4. Element's *Hosted Payment Window* collects (via swiped or keyed entry), stores (if applicable), transmits, and processes all sensitive cardholder data, and displays the results of the transaction within the window. In addition, the results of the transaction are embedded in the response window provided, thus providing synchronous knowledge of the transaction.
- 5. The POS can now programmatically query Element's real-time reporting API, **TransactionQuery**, using the *TransactionSetupID* for the specific purpose of obtaining the asynchronous result of the payment transaction. The **PaymentAccountQuery** method can be used to obtain details about records stored in *TransForm Tokenization*.

Hosted Payment Page (for web-based software applications)

The *Hosted Payment Page* essentially works the same way as the *Hosted Payment Window*. The below outlines the stept to process a *Hosted Payment Page* transaction with a typical Web-based POS:

- 1. The POS collects all of the non-sensitive data related to the purchase.
- The POS submits this non-sensitive data by calling the TransactionSetup method. A
 TransactionSetupID (GUID) is returned and used to identify the specific purchase from that point
 forward.
- The POS redirects from its site to Element's Hosted Payments system: https://certtransaction.hostedpayments.com/?TransactionSetupID=[InsertHere].
- 4. Element's Hosted Payment Page collects (via swiped or keyed entry), stores (if applicable), transmits, and processes all sensitive cardholder data, and then displays the results of the transaction on the payment page. In addition, a redirect link is provided which contains the results embedded in the query string, thus providing synchronous knowledge of the transaction. Response values embedded in the query string of the redirect link may include, but may not be limited to, the following:

Parameter Name	Values	Description
HostedPaymentStatus	"Complete" or "Cancelled"	Status of Hosted Payments request ("Error" returned if browser communication fails)
TransactionSetupID	TransactionSetupID	TransactionSetupID (GUID)
TransactionID	TransactionID	Numeric TransactionID

Parameter Name	Values	Description
ExpressResponseCode	ExpressResponseCode	Numeric response code
ExpressResponseMessage	ExpressResponseMessage	Text response message
AVSResponseCode	AVSResponseCode	AVS match, no match, or partial match code
CVVResponseCode	CVVResponseCode	CVV match or no match code
ApprovalNumber	ApprovalNumber	Approval code
LastFour	NNNN	Last four digits of card processed
ValidationCode	ValidationCode	Code used to validate original transaction
CardLogo	CardLogo	Payment brand. Possible values include: "Visa", "Mastercard", "Discover", "Amex", "Diners Club", "JCB", "Carte Blanche", "Other"
ApprovedAmount	ApprovedAmount	Approved amount (*depending on account type, ApprovedAmount may not be returned)
ServicesID	ServicesID	Numeric ServicesID
PaymentAccountID	PaymentAccountID	PaymentAccountID GUID
CommercialCard	CommercialCard	Commercial Card Response Code
ResponseCode	ResponseCode	identifying type of commercial card if merchant profile supports this functionality
TipAmount	TipAmount	Tip Amount is returned if entered into driverless terminal device

5. The POS can now programmatically query Element's real-time reporting API, *TransactionQuery*, using the *TransactionSetupID* for the specific purchase, to obtain the asynchronous result of the payment transaction. The *PaymentAccountQuery* method can be used to obtain details about records stored in *TransForm Tokenization*.

Notes

ReturnURL Redirect Query String Response

The name/value pair parameters appended to the *ReturnURL* during the final redirect response may include additional fields over time. Please code your redirect parsing application to properly allow for this scenario.

Validation Code Comparison for Fraud Prevention

The *Validation Code* should be captured as part of the response initiated by the *TransactionSetup* request, and it should be compared to the *Validation Code* that is returned from the appended values in the *ReturnURL* when redirecting back to the application after a successful *Hosted Payments* transaction.

- If the *Validation Code* values from both responses **match**, then it can be confirmed that the response data was returned directly from Element.
- If the *Validation Code* values from both responses **do not match**, this indicates that the response data is not accurate and the merchant should not proceed with providing a product or service to the customer.

TransactionSetupID Expiration

A Hosted Payments TransactionSetupID is valid for up to ten (10) minutes after its initial creation. Upon exceeding that time limitation, the TransactionSetupID will expire, and a new TransactionSetupID must be generated to complete the transaction.

CardInputCode

The CardInputCode enumerator within the TransactionSetup method identifies the payment transaction entry mode as either swiped or keyed entry. A CardInputCode enumerator value of MagstripeRead (2) will require a card to be swiped within the window or page, while all other values will require the card number and expiration date to be keyed within the window or page.

Entry Devices

Software applications that wish to support card-swiped entry within *Hosted Payments* are required to support an encrypted card reader or encrypted device provided by Element.

PIN-Debit

Hosted Payments supports PIN-Debit processing via encrypted PIN pad devices only.

Hosted Payments Transaction Support

Hosted Payments supports the following TransactionSetupMethod values at this time: CreditCardSale, CreditCardAuthorization, CreditCardAVSOnly, CreditCardForce, DebitCardSale, PaymentAccountCreate, PaymentAccountUpdate, Sale.

TransForm® Tokenization (Payment Account Secure Storage)

TransForm Tokenization is designed to provide merchants and payment service providers the ability to easily comply with PCI DSS with very little time, effort, or financial impact. Tokenization greatly reduces an enormous financial risk inherent with storing, processing, and/or transmitting cardholder data.

TransForm Tokenization is designed as a secure Web Service comprised of four basic methods. These methods are used to create, update, delete, and query payment data storage accounts:

Me	ethod	Description
1.	PaymentAccountCreate	The Create method inserts a new payment account record.
		See PaymentAccountCreate for details.
2.	PaymentAccountUpdate	The <i>Update</i> method amends an existing payment account record.
		See PaymentAccountUpdate for details.
3.	PaymentAccountDelete	The <i>Delete</i> method removes an existing payment account record.
		See PaymentAccountDelete for details.
4.	PaymentAccountQuery	The Query method retrieves an existing payment account record.
		See PaymentAccountQuery for details.

Tokenization Combined with End-to-End Encryption

TransForm Tokenization combines tokenization with end-to-end encryption. This allows developers to offload the huge responsibility of cardholder data storage to a trusted partner. Sensitive information for a customer sale is sent to *Express* only one time (using encryption) to create an initial account record and is then stored on OUR servers. A unique account identifier, called a **GUID**, is returned to the merchant and ALL subsequent sales for the customer refer to the *GUID*. In this way, risk is greatly reduced for all subsequent transactions because sensitive account numbers, address info, etc. never need be sent twice. The merchant does not have to store sensitive information AT ALL, and

implementing *TransForm Tokenization* is easy, inexpensive, fast, and reliable. The merchant is free to spend more time running their business and NOT worrying about the threat of data theft and resulting loss of customer confidence.

Processing TransForm Tokenization Records as Payments (Transaction)

To process a transaction using *TransForm Tokenization*, the method of the transaction (e.g. *CreditCardSale* or *CheckSale*) is called. However, instead of passing the card/account number and expiration information, the appropriate *PaymentAccountID* identifying the stored record is submitted. The transaction will be processed in the same way as any normal transaction. Billing address and zip code information will be submitted for AVS automatically if it is stored with the payment account token. Otherwise, any billing address and zip code sent separately in the request will override the stored values and will be submitted for AVS.

Updating TransForm Tokenization Records: PaymentAccountUpdate

When performing a *PaymentAccountUpdate* to update the card number or expiration date, each of these Card object fields, along with the full address information (e.g. Address object), must be resubmitted in the update request.

- If updating only an expiration date, the card number is not required.
- **To update an Address object field only,** the Card object fields are not required, but the full address details must be resubmitted in the update request.
- If all address details of the stored record are not known at that time, the *PaymentAccountQuery* method should be called first to collect this information.
- Note The PaymentAccountType value cannot be updated once the record is created.

Querying TransForm Tokenization Records: PaymentAccountQuery

When performing a *PaymentAccountQuery*, the search parameters include a credit card beginning and ending expiration date. This functionality allows merchants to search any range of card expiration date information so that currently expired and/or future expired cards can be handled appropriately.

Currently Supported Payment Types

Credit Cards and **Checks** can be processed using *TransForm Tokenization* technology. Note that *PaymentAccountID's* can be charged across multiple *AcceptorID's* as long as each *AcceptorID* is set up under the same corporate *AccountID* for the merchant.

Account Updater

What is Account Updater?

Element's Account Updater service provides merchants with the ability to receive automatic updates to existing records stored with TransForm Tokenization to prevent billing to expired or invalid cards. Depending on how the merchant is signed up to flag expired or invalid cards, the Account Updater will query the payment brands (Visa and MasterCard) for updated cardholder information. If the information is available, the customer's account on file (payment account billing record) will be automatically updated and available for future billing.

How does Account Updater work?

Merchant locations must first enroll in Element's *Account Updater* service, which is designed to work through a software application integration to the *Express* platform. Software applications can be set to

flag *TransForm Tokenization* records for update each time a stored payment account transaction is declined, and can also allow merchants to manually flag for update any stored payment account record of their choosing. At the end of the day, all cards that have been flagged will be submitted to the *Account Updater* service. Software applications can query flagged payment account records to determine if a card has been successfully updated. Updates are usually received within four to five business days. If an update was successful, merchants can submit the updated *TransForm Tokenization* record for future billing. Note that if the software application stores any non-sensitive, truncated card information, then that truncated information may also need to be updated to reflect the updated payment account record on file.

Which cards can be flagged for update?

Only Visa and MasterCard cards can be enrolled in the Account Updater service.

Which transaction declines should be flagged for update?

If manually flagging records, it is recommended that the following **ExpressResponseCode** decline values be flagged for update:

RC	Туре	Description
20	Decline	Card issuer has declined the transaction.
21	Expired Card	Card is expired.
24	Pick Up Card	Card issuer has declined the transaction and wants to recover the card.

Which updates are considered valid?

The following **PASSUpdaterStatus** response values resulting from a *PaymentAccountQuery* request are considered valid updates:

Match Type	Description
Match: Account Change	Account match was found and account number has changed. The <i>TransForm Tokenization</i> record has been updated with the new credit card number.
Match: Expiration Change	Account match was found and expiration date has changed. The TransForm Tokenization record has been updated with the new expiration date.
Match: Account Closed	Account match was found. Account has been closed. The <i>TransForm Tokenization</i> record will not be updated. This allows a merchant to identify <i>TransForm Tokenization</i> records that will never produce an approval by distinguishing between declines that are legitimate (due to closed accounts) and those resulting because the cardholder is too close to their credit limit to use the card on file.
Match: Contact Cardholder	Account match was found. The <i>TransForm Tokenization</i> record will not be updated. The merchant will need to contact the cardholder to inquire about the status of the account. This information allows a merchant to identify <i>TransForm Tokenization</i> records that will never produce an approval by distinguishing between declines that are legitimate (due to closed accounts) and those resulting because the cardholder is too close to the credit limit to use the card on file.

- To flag an existing TransForm Tokenization payment account record for a one-time manual update
 (recommended after specified declines), the software application can initiate a
 PaymentAccountAutoUpdate request using a PASSUpdaterBatchStatus of "IncludedInNextBatch".
- To flag an existing *TransForm Tokenization* record to be **updated automatically** after any future specified decline, the **PASSUpdaterOption** should be set to "AutoUpdateEnabled".
- For direct integrations (where PaymentAccountCreate is submitted directly from the software application), software applications can also flag a TransForm Tokenization record for a one-time manual update by submitting the PASSUpdaterBatchStatus of "IncludedInNextBatch" in the PaymentAccountCreate request.
- To determine the current status of a previously-flagged record, the software can initiate a PaymentAccountQuery request using PASSUpdaterStatus, PASSUpdaterBatchStatus, PASSUpdaterDateTimeBegin, PASSUpdaterDateTimeEnd, or any other existing query parameter (such as PaymentAccountID, PaymentAccountReferenceNumber, Expiration Date, PaymentBrand, etc.) A full list of possible update responses is shown below:

Match Type	Description	
Null	N/A	
Update in progress	Update in progress.	
Match: No Changes	Account match was found, but there are no changes.	
Match: Account Change	Account match was found and account number has changed. Account number has been updated.	
Match: Expiration Change	Account match was found and expiration date has changed. Expiration date has been updated.	
Match: Account Closed	Account match was found. Account has been closed.	
Match: Contact Cardholder	Account match was found. Contact the cardholder to update the account information.	
No Match: Participating	Account match not found on participating BIN.	
No Match: Non- Participating	Account match not found on non-participating BIN.	
Invalid Info	Invalid information. Example: Invalid card number.	
No Response	No response. Example: Request submitted, but not response.	
Not Allowed	Not allowed. Example: Merchant may not be set up for Account Updater service.	
Error	Error.	
PASS Updater Disabled	PASS Updater is disabled.	
Not Updated	Not updated.	

• To disable the ability for a currently-flagged record to be updated, the software application can submit a *PaymentAccountAutoUpdate* request using a *PASSUpdaterOption* of "AutoUpdateDisabled" and a *PASSUpdaterBatchStatus* of "NotIncludedInNextBatch".

Notes

- As long as the merchant is enrolled in the Account Updater service, the PaymentAccountCreate looks at the PASSUpdaterBatchStatus (e.g. IncludedInNextBatch) only. PaymentAccountCreate will ignore the PASSUpdaterOption (e.g. Disable).
- On all *PaymentAccountAutoUpdate* requests, the *PASSUpdaterOption* of *AutoUpdateDisabled* overrides any other existing status or value.

Security Methods and Procedures

Element Payment Services, Inc., as a financial service provider, is required to comply with the **Payment Card Industry (PCI) Data Security Standard (PCI DSS)**. The PCI Data Security Standard comprises twelve requirements pertaining to information security, including strong access controls, configuring security parameters appropriately on systems, tracking and monitoring data access, and restricting physical access to the systems containing card data.

IntegratingExpress with your software provides merchants with end-to-end encryption of cardholder data as well as the ability to store sensitive data off-site to meet PCI DSS compliance requirements.

Element Express requires all transactions to be sent using standard Secure Socket Layer (SSL) encryption. Additionally, every transaction must be authenticated. Authentication is achieved though use of an **AccountID** and **AccountToken** issued by Vantiv Integrated Payments. The full details of the PCI requirements can be found at http://www.visa.com/cisp.

Interface Specification

This section provides descriptions of common fields and extended parameters for **Healthcare**, **Auto Rental**, and **Lodging**.

Common Fields Descriptions

This section provides a reference list of fields most commonly used in *Express* methods and classes.

Input Fields

Name	Class	Required	Description
AcceptorID	Credentials	Required	MerchantID
AccountID	Credentials	Required	Unique account identifier
AccountNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Account Number of the consumers bank account.
AccountToken	<u>Credentials</u>	Required	Secret token used for authentication
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	Application	Required	Version of application
BillingAddress1	Address	Optional	The street address used for billing purposes
BillingZipcode	<u>Address</u>	Optional	The zip code used for billing purposes
CardNumber	<u>Card</u>	Conditional ¹	Account number on card
CardPresentCode	Terminal	Required	Card Present Code
© CAVV	<u>Card</u>	Optional	Card Authorization Verification Value
CheckNumber	<u>DemandDepositAccount</u>	Optional	Check Number
□ CheckType	<u>DemandDepositAccount</u>	Optional	Check Type (Personal or Business)
♥ CVV	<u>Card</u>	Optional	Card Verification Value: Visa CVV2, MasterCard CVC2, Discover CID
 ■ CVVPresenceCode	<u>Terminal</u>	Required	Card Verification Value Presence Code
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
MostBatchID	<u>Batch</u>	Conditional needed when BatchCloseType=Regular	Unique host batch identifier

Name	Class	Required	Description
■ MarketCode	Transaction	Required	
■ MotoECICode	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
PaymentAccountID	ExtParm.PaymentAccount	Optional	
ReferenceNumber	<u>Transaction</u>	Optional	Unique value assigned by the transaction sender
RoutingNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Routing Number of the consumers bank.
SalesTaxAmount	Transaction	Optional	Sales tax amount (use 0.00 for commercial card tax exempt transactions)
TerminalCapabilityCode	<u>Terminal</u>	Required	Terminal Capability Code
TerminalEnvironme ntCode	<u>Terminal</u>	Required	Terminal Environment Code
TerminalID	<u>Terminal</u>	Required	Unique terminal identifier
TicketNumber	Transaction	Optional	
▼ Track1Data	Card	Optional	
▼ Track2Data	Card	Optional	
TransactionAmount	<u>Transaction</u>	Required	Amount of transaction

Output Fields

Response Class

Name	Value	Max Length	Description
ApprovalNumber		30	Issuer assigned approval number
AVSResponseCode		2	Address Verification Service Response Code
CardLogo		20	Payment brand: Visa, MasterCard, Discover, Amex, Diners Club, JCB, Carte Blanche, Other
CVVResponseCode		2	Card Verification Value Response Code
ExpressResponseCode	Any valid <i>Express</i> response code	N/A	Express Response Code
ExpressResponseMessage	Any valid Express message	N/A	Express Response Message
ExpressTransactionDate		N/A	Express platform transaction date
ExpressTransactionTime		N/A	Express platform transaction time
•		N/A	Express platform transaction time

Name	Value	Max Length	Description
ExpressTransaction Timezone			zone
MostResponseCode	Any valid host response code	N/A	Host Response Code
HostResponseMessage	Any valid host response message	N/A	Host Response Message
HostTransactionDateTime		N/A	Host platform transaction date and time
QueryData	Any valid query data	N/A	Query information in xml format
ReportingData	Any valid reporting data	N/A	Transaction information in xml format
Reporting ID	Any valid <i>Express</i> reporting identifier	N/A	Unique reporting identifier
ServicesID	Any valid <i>Express</i> services identifier	N/A	Unique services identifier
TransactionID	Any valid <i>Express</i> transaction identifier	N/A	Unique transaction identifier

Healthcare Extended Parameter Fields

Used when submitting a Healthcare transaction (TSYS, FDC, or Global platform only).

Name	Class	Description
₽ HealthcareFlag	ExtParm.Healthcare	If this is set to True, the Healthcare parameters should be submitted.
HealthcareFirstAccountType	ExtParm.Healthcare	Healthcare account type
HealthcareFirstAmountType	ExtParm.Healthcare	Healthcare amount type
HealthcareFirstCurrencyCode	ExtParm.Healthcare	Currency Code. Use 840 for U.S.
■ HealthcareFirstAmountSign	ExtParm.Healthcare	Positive or negative amount
HealthcareFirstAmount	ExtParm.Healthcare	Healthcare amount
■ HealthcareSecondAccountType	ExtParm.Healthcare	Healthcare account type
■ HealthcareSecondAmountType	ExtParm.Healthcare	Healthcare amount type
HealthcareSecondCurrencyCode	ExtParm.Healthcare	Currency Code. Use 840 for U.S. Leave blank if not used.
HealthcareSecondAmountSign	ExtParm.Healthcare	Positive or negative amount
HealthcareSecondAmount	ExtParm.Healthcare	Healthcare amount. Leave blank if not used.
HealthcareThirdAccountType	ExtParm.Healthcare	Healthcare account type
HealthcareThirdAmountType	ExtParm.Healthcare	Healthcare amount type
* HealthcareThirdCurrencyCode	ExtParm.Healthcare	Currency Code. Use 840 for U.S. Leave blank if not used.
HealthcareThirdAmountSign	ExtParm.Healthcare	Positive or negative amount
HealthcareThirdAmount	ExtParm.Healthcare	Healthcare amount. Leave blank if not used.
HealthcareFourthAccountType	ExtParm.Healthcare	Healthcare account type
HealthcareFourthAmountType	ExtParm.Healthcare	Healthcare amount type

Name	Class	Description
HealthcareFourthCurrencyCode	ExtParm.Healthcare	Currency Code. Use 840 for U.S. Leave blank if not used.
HealthcareFourthAmountSign	ExtParm.Healthcare	Positive or negative amount
HealthcareFourthAmount	ExtParm.Healthcare	Healthcare amount. Leave blank if not used.

Auto Rental Extended Parameter Fields

Used when submitting an Auto Rental transaction (TSYS and FDC platform only).

Name	Class	Description
AutoRentalAgreementNumber	ExtParm.AutoRental	Rental agreement number
AutoRentalNoShowIndicator	ExtParm.AutoRental	No show indicator
AutoRentalExtraChargesDetail	ExtParm.AutoRental	Extra charges detail
AutoRentalPickupDate	ExtParm.AutoRental	Rental pickup date. Format is YYYYMMDD.
AutoRentalDropoffDate	ExtParm.AutoRental	Reserved for future use
AutoRentalCustomerName	ExtParm.AutoRental	Name of the person or business with the reservation
AutoRentalReturnCity	ExtParm.AutoRental	City where vehicle was returned
AutoRentalReturnState	ExtParm.AutoRental	State where vehicle was returned
AutoRentalReturnLocationID	ExtParm.AutoRental	Rental location ID
AutoRentalDuration	ExtParm.AutoRental	Number of rental days (e.g. 03). Pass 01 for noshow rentals.
AutoRentalPickupLocation	ExtParm.AutoRental	Name of business where vehicle was picked up
AutoRentalPickupCity	ExtParm.AutoRental	Name of city, town, or village where vehicle was picked up
AutoRentalPickupState	ExtParm.AutoRental	Abbreviation of state, province, or other country subdivision where vehicle was picked up
AutoRentalPickupCountryCode	ExtParm.AutoRental	Country code where vehicle was picked up. For US, use 840.
AutoRentalPickupTime	ExtParm.AutoRental	Local time when vehicle was picked up. Format HHMMSS.
AutoRentalReturnCountryCode	ExtParm.AutoRental	Country code where vehicle was returned. For US, use 840.
AutoRentalReturnDate	ExtParm.AutoRental	Date when vehicle was returned. Format YYYYMMDD.
AutoRentalReturnTime	ExtParm.AutoRental	Local time when vehicle was returned. Format HHMMSS.
AutoRentalVehicleClassCode	ExtParm.AutoRental	Classification of rental vehicle
AutoRentalDistance	ExtParm.AutoRental	Numeric distance traveled during rental period
¹ <u>AutoRentalDistanceUnit</u>	ExtParm.AutoRental	Unit of measurement applicable to distance traveled
AutoRentalAuditAdjustmentCode	ExtParm.AutoRental	Indicates if adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement
AutoRental Audit Adjustment Amount	ExtParm.AutoRental	Adjustment amount for mileage, fuel, vehicle damage, etc. after a vehicle was returned

Lodging Extended Parameter Fields

Used when submitting a Lodging transaction (TSYS and FDC platform only).

Name	Class	Description
LodgingAgreementNumber	ExtParm.Lodging	Hotel Folio number
LodgingCheckInDate	ExtParm.Lodging	Check-in date. Format is YYYYMMDD. Advance check-in = date of reservation No-show = anticipated arrival date
LodgingCheckOutDate	ExtParm.Lodging	Check-out date. Format is YYYYMMDD.
LodgingRoomAmount	ExtParm.Lodging	Nightly rate for one room
LodgingRoomTax	ExtParm.Lodging	Reserved for future use
LodgingNoShowIndicator	ExtParm.Lodging	Indicates a room was reserved, but not actually rented. 0 = Not Applicable 1 = No Show
LodgingDuration	ExtParm.Lodging	Number of hotel stay days (e.g. 03). Pass 01 for no-show. From 01 to 99.
LodgingCustomerName	ExtParm.Lodging	Lodging customer name
LodgingClientCode	ExtParm.Lodging	Reserved for future use
Lodging Extra Charges Detail	ExtParm.Lodging	Indicates any additional ancillary charges. Field can contain multiple indicators. 0 = Not Used 1 = Reserved 2 = Restaurant 3 = Gift Shop 4 = Mini-Bar 5 = Telephone 6 = Other 7 = Laundry
LodgingExtraChargesAmounts	ExtParm.Lodging	Reserved for future use
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Prestigious property code
	ExtParm.Lodging	Special program code
LodgingChargeType	ExtParm.Lodging	Lodging charge type

EnhancedData Extended Parameter Fields

Used when submitting a Level III transaction (TSYS, FDC, and Vantiv platforms only).

Note Only American Express Level III items are supported on Vantiv platform.

Name	Class	Max	Required	Description
MerchantVATRegistrationNumber	ExtendedParameters EnhancedData	20	Optional ¹	Government-assigned tax identification number of merchant. (tag must be present, but can be empty)
CustomerVATRegistrationNumber	ExtendedParameters EnhancedData	13	Optional ¹	Tax registration number supplied by cardholder. (tag must be present, but can be empty)
	ExtendedParameters EnhancedData	4	Required ¹	International description code of overall goods or services.
DiscountAmount	ExtendedParameters EnhancedData	12	Required ¹	Discount amount applied to the line item total. This field should be 0.00 if the unit cost reflects a discount price or if no line item discount applies. DDDDDDDDDD.CC
FreightAmount	ExtendedParameters EnhancedData	12	Required ¹	Freight or shipping portion of the total transaction amount. DDDDDDDDDDD.CC
DutyAmount	ExtendedParameters EnhancedData	12	Required ¹	Fee amount associated with the import of the goods. DDDDDDDDDD.CC
◆ DestinationZIPCode	ExtendedParameters EnhancedData	10	Required ¹	Postal/Zip Code of the address where purchased goods will be delivered. Can be the same as ShipFromZIPCode if customer is present and takes immediate possession of the goods.
ShipFromZIPCode	ExtendedParameters EnhancedData	10	Required ¹	Postal/Zip Code of the address from where the purchased goods are being shipped.
DestinationCountryCode	ExtendedParameters EnhancedData	3	Required ¹	Country code where goods are being shipped. U.S. is 840.
UniqueVATInvoiceReference Number	ExtendedParameters EnhancedData	15	Optional ¹	Invoice number associated with VAT invoice. Not applicable for US merchants. (tag must be present, but can be empty)
OrderDate	ExtendedParameters EnhancedData	8	Required ¹	Purchase order date. YYYYMMDD

Name	Class	Max	Required	Description
	ExtendedParameters EnhancedData	12	Required ¹	Amount of any sales or value added taxes associated with the purchased item. DDDDDDDDDD.CC
VATRate	ExtendedParameters EnhancedData	4	Required ¹	Tax rate used to calculate VATAmount. Submitted as a percentage. Two-digit implied decimal. For example, if 10.42%, submit 1042
LineItemCount	ExtendedParameters EnhancedData	3	Required ¹	Number of line items included in transaction. Visa (0 to 998), MasterCard (0 to 98)
AlternateTaxAmount	Extended Parameters Enhanced Data	12	Required ¹	Second tax amount (used in countries where more than one tax can be applied). Not applicable for US merchants. DDDDDDDDDDD.CC
NationalTaxAmount	ExtendedParameters EnhancedData	12	Required ¹	National tax included in the transaction amount. Not applicable for US merchants. DDDDDDDDDDD.CC
RequesterName	ExtendedParameters EnhancedData	38	Conditional ¹ (for Amex Level III)	Name of the requester if the card member is not the person that ordered the product/service
RequesterCityName	ExtendedParameters EnhancedData	20	Conditional 1 (for Amex Level III)	Requester city name
RequesterStateProvinceCode	Extended Parameters Enhanced Data	2	Conditional ¹ (for Amex Level III)	Requester state or province code
RequesterPostalCode	ExtendedParameters EnhancedData	15	Conditional 1 (for Amex Level III)	Requester postal code
RequesterCountryCode	Extended Parameters Enhanced Data	3	Conditional 1 (for Amex Level III)	Requester country code ("840")
	ExtendedParameters EnhancedData	12	Required ¹	International description code of the individual good or service being supplied. Recommend using national or international list of standardized codes.

Name	Class	Max	Required	Description
ItemDescription	ExtendedParameters	35	Required ¹	Item description.
	EnhancedData			
ProductCode	ExtendedParameters	12	Required ¹	Merchant-defined description
	EnhancedData			code of the item.
Quantity	ExtendedParameters	12	Required ¹	Quantity of the item (up to four
	EnhancedData			decimal places). DDDDDDD.CCCC
UnitOfMeasure	ExtendedParameters	12	Required ¹	Code for units of measurement used in international trade.
	EnhancedData			useu III IIIterriational traue.
UnitCost	ExtendedParameters	10	Required ¹	Unit cost of the item. DDDDDDD.CC
	EnhancedData			
LineItemVATAmount	ExtendedParameters	12	Required ¹	Amount of any sales or value added taxes associated with
	EnhancedData			the item.
LineItemVATRate	ExtendedParameters	4	Required ¹	Tax rate used to calculate
	EnhancedData			LineItemVATAmount. Submitted as a percentage. Two-digit implied decimal. For example, if 10.42%, submit 1042
LineItemDiscountAmount	ExtendedParameters	12	Required ¹	Discount amount applied to the
	EnhancedData			line item. This field should be 0.00 if the unit cost reflects a discount price or if no line item discount applies. DDDDDDDDDD.CC
LineItemTotalAmount	ExtendedParameters	12	Required ¹	Total full amount associated
	EnhancedData			with the line item (including tax). DDDDDDDDD.CC
Alternate Tax I dentifier	ExtendedParameters	15	Optional ¹	Tax identification number of
	EnhancedData			the merchant reporting the alternate tax amount. Not applicable for US merchants. (tag must be present, but can be empty)
₽ VATType	ExtendedParameters	4	Optional ¹	Type of value added taxes
	EnhancedData			being used. Not applicable for US merchants. (tag must be present, but can be empty)
■ DiscountCode	ExtendedParameters	2	Required ¹	Indicates whether discount was
	EnhancedData			applied to the purchase price of the item.
				0 = Not supported

Name	Class	Max	Required	Description
				1 = Amount is discounted
				2 = Amount is not discounted
^{■P}NetGrossCode	ExtendedParameters EnhancedData	2	Required ¹	Indicates whether extended item amount included the tax.
	2			0 = Not supported
				1 = Item Amount includes tax amount
				2 = Item Amount does not include tax amount
ExtendedItemAmount	ExtendedParameters	12	Required ¹	Amount of the item (price
	EnhancedData			multiplied by quantity). DDDDDDDDDD.CC
 ■ PDebitCreditCode	ExtendedParameters	2	Required ¹	Indicates whether
	EnhancedData			ExtendedItemAmount is a Credit (taking away from the total amount of the sale) or Debit (adding to the total amount of the sale).
				0 = Not supported
				1 = Extended Item Amount is Credit
				2 = Extended Item Amount is Debit
ItemDiscountRate	ExtendedParameters	5	Required ¹	Discount rate for the line item.
	EnhancedData			Submitted as a percentage. Two-digit implied decimal. For example, if 10.42%, submit 1042

Level III XML Format (Note that EnhancedData and LineItemDetail MUST be in this order):

```
<ExtendedParameters>
 <EnhancedData>
    <MerchantVATRegistrationNumber />
    <CustomerVATRegistrationNumber />
    <SummaryCommodityCode />
    <DiscountAmount />
    <FreightAmount />
    <DutyAmount />
    <DestinationZIPCode />
    <ShipFromZIPCode />
    <DestinationCountryCode />
    <UniqueVATInvoiceReferenceNumber />
    <OrderDate />
    <VATAmount />
    <VATRate />
    <LineItemCount />
    <AlternateTaxAmount />
    <NationalTaxAmount />
<RequesterName />
<RequesterCityName />
<RequesterStateProvinceCode />
<RequesterPostalCode />
<RequesterCountryCode />
    <LineItemDetail>
      <LineItem>
        <ItemCommodityCode />
        <ItemDescription />
        <ProductCode />
        <Quantity />
        <UnitOfMeasure />
        <UnitCost />
        <LineItemVATAmount />
        <LineItemVATRate />
        <LineItemDiscountAmount />
        <LineItemTotalAmount />
        <AlternateTaxIdentifier />
        <VATType />
        <DiscountCode />
        <NetGrossCode />
        <ExtendedItemAmount />
        <DebitCreditCode />
        <ItemDiscountRate />
      </LineItem>
      <LineItem>
      </LineItem>
    </LineItemDetail>
 </EnhancedData>
</ExtendedParameters>
```

Functionality

Duplicate Checking

Duplicate Checking is a host-based feature that can be turned on or off per transaction. It provides a means to identify duplicate transactions by the following parameters:

- Same Card Number
- Same Transaction Amount
- Specific Transaction Category (Sale, Return, etc.)
- Within a certain number of Transactions (25) in the same batch

There are two methods that can be used to either override or disable duplicate checking:

- DuplicateOverrideFlag: Set this flag to true to override the duplicate checking feature if the
 response from the host indicates a duplicate transaction has been attempted. In other words, if you
 send a transaction that was determined to be a duplicate by the host, set this flag to true and resend
 the same transaction to bypass the duplicate checking feature on the host. This is a per transaction
 request feature.
- **DuplicateCheckDisableFlag:** Set this flag to **true** to disable the duplicate checking feature on the host. If this flag is set to *true* no duplicate checks will be performed for that transaction. This is a per transaction request feature.

Recurring Transactions

Recurring transactions are a unique type of transaction where consumers authorize merchants or solution providers to bill a specific card on a regular basis (e.g. monthly membership fees). Each recurring transaction may be for the same or fluctuating dollar amounts. To perform recurring transactions merchants must store cardholder information. Cardholder information can either be stored locally or remotely using Element's *TransForm Tokenization* technology.

Note Storing cardholder information locally is a significant financial risk and is subject to the Payment Card Industry Data Security Standards (PCI DSS). Please visit www.pcisecuritystandards.org for more information.

Setting up recurring transactions consists of two parts:

Part 1: Set-Up Transaction: For the first transaction in the series, send in the same data as a normal sale transaction. This is Track Data for card present transactions, or Card Number, CVV, and AVS data for eCommerce and MOTO transactions.

Part 2: Recurring Transactions: For all subsequent recurring transactions, send in the cardholder account number (*not* Track Data) as well as Billing Street and Zip Code data, and set the RecurringFlag to "true."

Note PCI DSS prohibits the storing of Track Data and CVV data. Please visit www.pcisecuritystandards.org for more information.

Scheduled Tasks

CreditCardSale and **CheckSale** are the only methods that can be scheduled.

Scheduling a Task (Transaction)

To schedule a transaction, call the creation method of that transaction as you would normally do. The only difference is you will also pass in the ScheduledTask object in the ExtendedParameters array of that method. By doing this, Express will schedule the transaction.

Note Express will only schedule the transaction, it will not execute immediately.

When scheduling payment transactions, you may pass a *PaymentAccountID* for the payment information if you are already using *TransForm Tokenization* for storing sensitive cardholder or payment information. If you choose to pass in the original card or payment information, *Express* will automatically create a <u>PaymentAccount</u> (*TransForm Tokenization*) for you and store your payment data. Once the Scheduled Task is created, you will be returned both a Scheduled Task ID as well as a Payment Account ID.

Scheduling Tasks with varying parameters

You may have a scenario where you want to vary the parameters on each method call, such as charging a cardholder a different amount each month. In order to accomplish this, you would create multiple Scheduled Tasks with a Run Frequency of "OneTimeFuture", creating one task for each occurrence. These tasks can be grouped together for reporting purposes by assigning the same ScheduledTaskReferenceNumber to all.

Editing / Updating a Scheduled Task

The only items that you can update on a Scheduled Task are the members of the *ScheduledTask* class. You cannot update the actual method call parameters due to the complexity of parameters that get passed into the method. If you wish to modify those, you must delete the task and create a new one with the appropriate parameters.

Reporting on the results of Scheduled Tasks

The *ScheduledTaskQuery* method can be used to report on the criteria in place for a particular task, as well as report on transactions that have already run based on that task.

Check Processing

Determining the status of a Check transaction

Once created using the *CheckSale* method, the status of a check transaction can be determined by submitting a *TransactionQuery* request through the *Express* reporting interface. The check status will be returned in the *TransactionStatus* response field, and this status value will determine any action(s) that can be taken on the check transaction at that point. Due to the nature of ACH processing, the status of ACH transactions will be updated over a period of several days (and can be longer for transactions that are returned by the bank). Your software application must be prepared to frequently update existing records.

Voiding a Check transaction

The ability to void a check transaction is dependent on the status of that check transaction. Check transactions with a *TransactionStatus* of "Pending" can be voided using the *CheckVoid* method. Attempting a *CheckVoid* on a check transaction in any other status may result in an error response. After successfully processing a check transaction, the status may be in a "Pending" state until daily back-office processing takes place.

Reversing a Check transaction

Reversing a check transaction is dependent on the status. Check transactions with a *TransactionStatus* of "Pending" can be reversed using the *CheckReversal* method. Attempting a *CheckReversal* on a check transaction in any other status may result in an error response. After successfully processing a check transaction, the status may be in a "Pending" state until daily back-office processing takes place. Another scenario requiring a *CheckReversal* would be in the event that no response is received from the original check transaction. In that case, a *CheckReversal* should be submitted. If *Express* cannot find the *TransactionID* submitted for the original check transaction, a "Transaction cannot be reversed" response will be returned.

Refunding a Check transaction (CheckReturn)

The ability to refund a check transaction is dependent on the status of that check transaction. Check transactions with a *TransactionStatus* of "Originated" or "Settled" can be refunded using the *CheckReturn* method. Attempting a *CheckReturn* on a check transaction in any other status may result in an error response. At this time, the refund amount must be equal to the original sale amount. If a refund for an amount other than the original sale amount is required, the *CheckCredit* method should be used.

Level III Processing

Level III Line Item Detail

Express supports the ability to process Visa and MasterCard Level III Enhanced and Line Item Detail. American Express Level III data is supported on Vantiv platform only.

Processing Flow

Sale Transactions (charge immediately)

- The software application will submit a standard CreditCardSale request (without any Level III information). If the card processed is a commercial card, a response value called CommercialCardResponseCode may be returned indicating the card is a type of commercial card. This will identify the commercial card as a Business Card, Corporate Card, or Purchasing Card.
- 2. Where applicable and when Level III information is available, a CreditCardAdjustment request (containing Level III Enhanced and Line Item Details) will be submitted to adjust the original CreditCardSale to include Level III information for settlement. ALL Level III Enhanced and Line Item Details are required when including this Level III Data.
- 3. Sale transaction batches out and settles automatically during the overnight hours.

Authorization and Authorization Completion Transactions (charge when shipped)

- 1. The software application will submit a standard *CreditCardAuthorization* request (without any Level III information). If the card processed is a commercial card, a response value called *CommercialCardResponseCode* may be returned indicating the card is a type of commercial card. This will identify the commercial card as a Business Card, Corporate Card, or Purchasing Card.
- 2. Where applicable and when Level III information is available, a *CreditCardAuthorizationCompletion* request (containing Level III Enhanced and Line Item Details) will be submitted to complete the original *CreditCardAuthorization* to include Level III information for settlement. *ALL* Level III Enhanced and Line Item Details are required when including this Level III Data.
- 3. Completion transaction batches out and settles automatically during the overnight hours.

Force Transactions (charge immediately)

- Upon receiving a voice authorization response, the software application will submit a standard CreditCardForce request (without any Level III information). Depending on the information known about the card, the software may choose to include Level III details by generating a CreditCardAdjustment request to include Level III details as a secondary step in the process. Whenever provided, ALL Level III details are required when including this Enhanced Data.
- 2. Where applicable and when Level III information is available, a *CreditCardAdjustment* request (containing Level III Enhanced and Line Item Details) will be submitted to adjust the original *CreditCardForce* to include Level III information for settlement. *ALL* Level III Enhanced and Line Item Details are required when including this Level III Data.
- 3. Force transaction batches out and settles automatically during the overnight hours.

Level III Notes

- Level III Enhanced and Line Item Details are supported via XML interface only (not available via SOAP interface).
- Level III Enhanced and Line Item Details are supported via TSYS, FDC, and Vantiv processing platforms only.
- Level III Enhanced and Line Item Details are supported for Visa and MasterCard transactions. American Express Level III data is supported on Vantiv platform only.
- Level III Enhanced and Line Item Details have a limit on the number of line items included in a single transaction.
- Where necessary (and based on the CommercialCardResponseCode), the CreditCardAdjustment method can be used to update original CreditCardSale, CreditCardAuthorizationCompletion, or CreditCardForce requests to include Level III Details.

Methods

Batch

BatchClose

The **BatchClose** method is used to close a currently active processing batch. It may or may not be needed depending on how your merchant account is set up. Merchants set up as **Merchant-Initiated Batch Close** will need to close their own batches. If you are set up as **Time-Initiated Batch Close** you do not need to use this method.

There are two different *BatchCloseType* settings you can use:

- Regular: This is the most common. Before calling BatchClose with this BatchCloseType, you will
 need to know the totals of your batch. These can come from your own totals, or you can call the
 BatchTotalsQuery method to get the totals you need. You may want to consider calling
 BatchTotalsQuery first and then compare your totals to the host's before attempting to call
 BatchClose.
- **Force**: If you call *BatchClose* with this *BatchCloseType*, the host batch will be closed without the user sending in any totals.

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
* AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
* AcceptorID	Credentials	Required	MerchantID
* TerminalID	Terminal	Required	Unique terminal identifier
BatchCloseType	<u>Batch</u>	Required	Type of batch close
HostBatchID	<u>Batch</u>	Required	Unique host batch identifier
HostBatchCount	<u>Batch</u>	Conditional	Batch count
		needed when BatchCloseType=Regular	
HostCreditSaleCount	<u>Batch</u>	Conditional	Total numeric count of credit sale items
		needed when BatchCloseType=Regular	credit sale items
HostCreditSaleAmount	<u>Batch</u>	Conditional	Total dollar amount of credit
	nee Bato		sales with decimal place: example: 21.52

Name	Class	Required	Description
HostCreditReturnCount	Batch	Conditional	Total numeric count of
		needed when BatchCloseType=Regular	credit return items
HostCreditReturnAmount	<u>Batch</u>	Conditional	Total dollar amount of credit
		needed when BatchCloseType=Regular	returns with decimal place: example: 21.52
HostDebitSaleCount	<u>Batch</u>	Conditional	Total numeric count of debit
		needed when BatchCloseType=Regular	sale items
HostDebitSaleAmount	<u>Batch</u>	Conditional	Total dollar amount of debit
		needed when BatchCloseType=Regular	sales with decimal place: example: 21.52
HostDebitReturnCount	Batch	Conditional	Total numeric count of debit
		needed when BatchCloseType=Regular	return items
HostDebitReturnAmount	Batch	Conditional	Total dollar amount of debit
		needed when BatchCloseType=Regular	returns with decimal place: example: 21.52

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
* HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
* TransactionID	Response.Transaction	Returned	Unique transaction identifier
ProcessorName	Response.Transaction	Returned	Name of processor
▼ <u>TransactionStatus</u>	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
HostBatchID	Response.Batch	Returned	Unique host batch identifier

Name	Class	Returned	Description
MostBatchCount HostBatchCount	Response.Batch	Returned	Total numeric count of all items in the Batch
MostBatchAmount Page 1	Response Batch	Returned	Total dollar amount of all items
MostCreditSaleCount	Response.Batch	Returned	Total numeric count of credit sale items
MostCreditSaleAmount	Response.Batch	Returned	Total dollar amount of credit sales
MostCreditReturnCount	Response.Batch	Returned	Total numeric count of credit return items
MostCreditReturnAmount	Response.Batch	Returned	Total dollar amount of credit returns
MostDebitSaleCount	Response.Batch	Returned	Total numeric count of debit sale items
MostDebitSaleAmount	Response.Batch	Returned	Total dollar amount of debit sales
MostDebitReturnCount	Response.Batch	Returned	Total numeric count of debit return items
HostDebitReturnAmount	Response.Batch	Returned	Total dollar amount of debit returns

☞Notes for closing a batch

Batch details are available through the *BatchTotalsQuery* request method. When manually closing a batch via the *BatchClose* method, it is recommended that a *BatchCloseType* of **Force** be submitted.

■ BatchItemQuery

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	Application	Required	Version of application
AccountID AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
▼ TerminalID	Terminal	Required	Unique terminal identifier
BatchQueryType	<u>Batch</u>	Required	Type of batch query
MostBatchID HostBatchID	<u>Batch</u>	Optional	If null, current batch is used
MostItemID HostItemID	<u>Batch</u>	Optional	If null, last item sent in is returned

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
₱ HostResponseCode	Response	Returned	Host Response Code
₱ HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
MostBatchID HostBatchID	Response.Batch	Returned	Unique host batch identifier
MostItemID Property	Response.Batch	Returned	Unique host item identifier
ApprovalNumber	Response.Transaction	Returned	Total numeric count of credit sales
TransactionAmount	Response.Transaction	Returned	Amount of transaction
▼ TransactionID ▼ TransactionID	Response.Transaction	Returned	Unique transaction identifier
ProcessorName	Response.Transaction	Returned	Name of processor

Name	Class	Returned	Description
* TransactionStatus	Response Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction

Notes

• BatchItemQuery only queries against the current open batch.

BatchTotalsQuery

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
* AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
* AcceptorID	Credentials	Required	MerchantID
* TerminalID	Terminal	Required	Unique terminal identifier
■ BatchQueryType	<u>Batch</u>	Required	Identifies a Totals or Item query
MostBatchID	<u>Batch</u>	Optional	If null, current open batch is used
■ BatchIndexCode	<u>Batch</u>	Optional	Identifies to use current or previous batches.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
₱ HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
Express Transaction Date	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
* TransactionID	Response Transaction	Returned	Unique transaction identifier
ProcessorName	Response.Transaction	Returned	Name of processor
* TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
MostBatchID HostBatchID	Response.Batch	Returned	Unique host batch identifier
MostBatchCount	Response.Batch	Returned	Total numeric count of all items in the Batch
MostBatchAmount	Response.Batch	Returned	Total dollar amount of all items
MostCreditSaleCount	Response.Batch	Returned	Total numeric count of credit sale items
MostCreditSaleAmount	Response Batch	Returned	Total dollar amount of debit returns with decimal place: example: 21.52
HostCreditReturnCount	Response.Batch	Returned	Total numeric count of credit return items
MostCreditReturnAmount	Response Batch	Returned	Total dollar amount of debit returns with decimal place: example: 21.52
MostDebitSaleCount	Response.Batch	Returned	Total numeric count of debit sale items
* Host Debit Sale Amount	Response.Batch	Returned	Total dollar amount of debit returns with decimal place: example: 21.52
MostDebitReturnCount	Response.Batch	Returned	Total numeric count of debit return items

Name	Class	Returned	Description
MostDebitReturnAmount	Response.Batch	Returned	Total dollar amount of debit returns with decimal place: example: 21.52

Notes

- BatchTotalsQuery will only return complete totals for the current open batch.
- The grand total of a previous batch is available. If you wish to get the grand total for the previous batch, set the *BatchIndexCode* to "FirstPrevious."

Credit

CreditCardSale

The **Sale** transaction is the most basic and most common transaction, used for the sale/purchase of goods or services. This type of transaction compares the cardholder's available credit to the amount specified in the transaction. If the amount specified is available, the transaction is approved by the card issuer and an approval code is returned. If the amount is **not** available, or the transaction is denied for any other reason by the issuer, an error message containing the reason is returned.

Name	Class	Required	Description
* ApplicationID	Application	Required	Unique application identifier
* ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
* AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
* AcceptorID	Credentials	Required	MerchantID
* TransactionAmount	<u>Transaction</u>	Required	Amount of transaction. Depending on the merchant setup, a \$0.00 account verification may be used when address verification is performed.
© ClerkNumber	Transaction	Optional	Clerk number
♦ ShiftID • Shi	Transaction	Optional	Shift ID
SalesTaxAmount	<u>Transaction</u>	Optional	Sales tax amount. For Level II, submit amount for tax or 0.00 for tax-exempt.
TipAmount	Transaction	Optional	Tip amount
© ConvenienceFeeAmount	Transaction	Optional	Convenience fee amount (Vantiv only)
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Optional ³	Unique numeric value assigned by the transaction sender
CommercialCardCustomerCode	Transaction	Optional	Commercial Card Customer Code
<u>■ MarketCode</u>	Transaction	Required	Market Code
■ PartialApprovedFlag	Transaction	Conditional	Indicates whether software supports partial approvals

Name	Class	Required	Description
MerchantVerificationValue	Transaction	Conditional ⁵	Healthcare Merchant Verification Value
[©] DuplicateOverrideFlag	<u>Transaction</u>	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
* TerminalID	Terminal	Required	Unique terminal identifier
<u>E¹CardPresentCode</u>	<u>Terminal</u>	Required	Card Present Code
<u>E-CardholderPresentCode</u>	Terminal	Required	Cardholder Present Code
 <u>CardInputCode</u>	Terminal	Required	Card Input Code
© CVVPresenceCode	Terminal	Required	Card Verification Value Presence Code
<u>FTerminalCapabilityCode</u>	Terminal	Required	Terminal Capability Code
<u>FTerminalEnvironmentCode</u>	<u>Terminal</u>	Required	Terminal Environment Code
<u>■ MotoECICode</u>	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
<u>F</u> TerminalType	<u>Terminal</u>	Optional	Terminal Type
<u>FarminalEncryptionFormat</u>	<u>Terminal</u>	Optional	Not used
TerminalSerialNumber	<u>Terminal</u>	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	<u>Terminal</u>	Conditional ⁸	Lane Number
PaymentAccountID	ExtParm.PaymentAccount	Optional	Payment Account ID
Track1Data	Card	Conditional ¹	Track 1 data
Track2Data	<u>Card</u>	Conditional ¹	Track 2 data
MagneprintData MagneprintData	<u>Card</u>	Conditional ¹	Magneprint data
CardNumber	<u>Card</u>	Conditional ¹	Account number on card
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
EncryptedTrack1Data	Card	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	Card	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	Card	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption

Name	Class	Required	Description
			device
CardDataKeySerialNumber	Card	Conditional ¹	Card Data Key Serial Number from integrated encryption device
EncryptedFormat	Card	Optional	Encryption format from integrated encryption device
	<u>Card</u>	Optional	Card Verification Value:
			Visa CVV2, MasterCard CVC2, Discover CID
P CAVV	Card	Optional	Card Authorization Verification Value
× XID	Card	Optional	Verified by Visa transaction identifier
BillingName	Address	Optional	The name used for billing purposes. Recommend submitting for Level II processing.
BillingAddress1	Address	Conditional Needed for AVS	The street address used for billing purposes
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	Address	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	<u>Address</u>	Conditional Needed for AVS	The zip code used for billing purposes
BillingEmail	Address	Optional	The email address used for billing purposes
BillingPhone	Address	Optional	The phone number used for billing purposes
₱ ShippingName ■ Compare the state of	Address	Optional	The name used for billing purposes
ShippingAddress1	Address	Optional	The street address used for Shipping purposes
ShippingAddress2	Address	Optional	The street address used for Shipping purposes
ShippingCity	Address	Optional	The city name used for Shipping purposes

Name	Class	Required	Description
[♥] ShippingState	Address	Optional	The state name used for Shipping purposes
ShippingZipcode	Address	Optional	The zip code used for Shipping purposes. Recommend submitting for Level II processing.
ShippingEmail	Address	Optional	The email address used for Shipping purposes
ShippingPhone	Address	Optional	The phone number used for Shipping purposes
ScheduledTaskReferenceNumber	ExtParm.ScheduledTask	Optional ²	User defined unique reference number that identifies this scheduled task.
^{©P} RunFrequency	ExtParm.ScheduledTask	Optional ²	The frequency you want to task to run.
RunStartDate	ExtParm_ScheduledTask	Optional ²	The date you want the task to start.
^P RunUntilCancelFlag	ExtParm.ScheduledTask	Optional ²	A flag stating if this task should keep running at the selected Run Frequency until the user cancel it, or to just run for the given amount of RunCycles.
RunCycles	ExtParm_ScheduledTask	Optional ²	The amount of times you want the task to run.
[■] ScheduledTaskStatus	ExtParm. Scheduled Task	Optional ²	Active or Disabled
ScheduledTaskName	ExtParm.ScheduledTask	Optional ²	User generated name of the task.
ScheduledTaskGroupID	ExtParm.ScheduledTask	Optional ⁴	Express platform generates this.
i HealthcareFlag	ExtParm.Healthcare	Conditional ⁵	If this is set to True, the Healthcare parameters should be submitted.
■ HealthcareFirstAccountType	ExtParm.Healthcare	Conditional ⁵	Healthcare account type
HealthcareFirstAmountType	ExtParm.Healthcare	Conditional ⁵	Healthcare amount type
<u>HealthcareFirstCurrencyCode</u>	ExtParm.Healthcare	Conditional ⁵	Currency Code. Use 840 for U.S.
HealthcareFirstAmountSign	ExtParm.Healthcare	Conditional ⁵	Positive or negative amount
HealthcareFirstAmount	ExtParm.Healthcare	Conditional ⁵	Healthcare amount
HealthcareSecondAccountType	ExtParm.Healthcare	Conditional ⁵	Healthcare account type
■ HealthcareSecondAmountType	ExtParm.Healthcare	Conditional ⁵	Healthcare amount type
MealthcareSecondCurrencyCode	ExtParm.Healthcare	Conditional ⁵	Currency Code. Use 840 for U.S.

Name	Class	Required	Description
HealthcareSecondAmountSign	ExtParm.Healthcare	Conditional ⁵	Positive or negative amount
HealthcareSecondAmount	ExtParm.Healthcare	Conditional ⁵	Healthcare amount
HealthcareThirdAccountType	ExtParm.Healthcare	Conditional ⁵	Healthcare account type
HealthcareThirdAmountType	ExtParm.Healthcare	Conditional ⁵	Healthcare amount type
HealthcareThirdCurrencyCode	ExtParm.Healthcare	Conditional ⁵	Currency Code. Use 840 for U.S.
HealthcareThirdAmountSign	ExtParm.Healthcare	Conditional ⁵	Positive or negative amount
₱ HealthcareThirdAmount	ExtParm.Healthcare	Conditional ⁵	Healthcare amount
HealthcareFourthAccountType	ExtParm.Healthcare	Conditional ⁵	Healthcare account type
HealthcareFourthAmountType	ExtParm.Healthcare	Conditional ⁵	Healthcare amount type
₱ HealthcareFourthCurrencyCode	ExtParm.Healthcare	Conditional ⁵	Currency Code. Use 840 for U.S.
HealthcareFourthAmountSign	ExtParm.Healthcare	Conditional ⁵	Positive or negative amount
* HealthcareFourthAmount	ExtParm.Healthcare	Conditional ⁵	Healthcare amount
AutoRentalAgreementNumber	ExtParm.AutoRental	Conditional ⁶	Rental agreement number
AutoRentalNoShowIndicator	ExtParm.AutoRental	Conditional ⁶	No show indicator
AutoRentalExtraChargesDetail	ExtParm.AutoRental	Conditional ⁶	Extra charges detail
AutoRentalPickupDate	ExtParm.AutoRental	Conditional ⁶	Rental pickup date. Format is YYYYMMDD.
AutoRentalCustomerName	ExtParm.AutoRental	Conditional ⁶	Name of the person or business with the reservation
AutoRentalReturnCity	ExtParm.AutoRental	Conditional ⁶	City where vehicle was returned
AutoRentalReturnState	ExtParm.AutoRental	Conditional ⁶	State where vehicle was returned
AutoRentalReturnLocationID	ExtParm.AutoRental	Conditional ⁶	Rental location ID
AutoRentalDuration	ExtParm.AutoRental	Conditional ⁶	Number of rental days (e.g. 03). Pass 01 for no-show rentals.
AutoRentalPickupLocation	ExtParm.AutoRental	Conditional ⁶	Name of business where vehicle was picked up
AutoRentalPickupCity	ExtParm.AutoRental	Conditional ⁶	Name of city, town, or village where vehicle was picked up
AutoRentalPickupState	ExtParm.AutoRental	Conditional ⁶	Abbreviation of state, province, or other country subdivision where vehicle was picked up
AutoRentalPickupCountryCode	ExtParm.AutoRental	Conditional ⁶	Country code where vehicle was picked up. For US, use 840.
AutoRentalPickupTime	ExtParm.AutoRental	Conditional ⁶	Local time when vehicle was picked up.
AutoRentalReturnCountryCode	ExtParm.AutoRental	Conditional ⁶	Country code where vehicle was

Name	Class	Required	Description
			returned. For US, use 840.
AutoRentalReturnDate	ExtParm.AutoRental	Conditional ⁶	Date when vehicle was returned. Format YYYYMMDD.
AutoRentalReturnTime	ExtParm.AutoRental	Conditional ⁶	Local time when vehicle was returned
<u>AutoRentalVehicleClassCode</u>	ExtParm.AutoRental	Conditional ⁶	Classification of rental vehicle
* AutoRentalDistance	ExtParm.AutoRental	Conditional ⁶	Distance traveled during rental period
<u>AutoRentalDistanceUnit</u>	ExtParm.AutoRental	Conditional ⁶	Unit of measurement applicable to distance traveled
AutoRentalAuditAdjustmentCode	ExtParm.AutoRental	Conditional ⁶	Indicates if adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement
AutoRentalAuditAdjustmentAmount	ExtParm.AutoRental	Conditional ⁶	Adjustment amount for mileage, fuel, vehicle damage, etc. after a vehicle was returned
LodgingAgreementNumber	ExtParm.Lodging	Conditional ⁷	Agreement (Folio) number
LodgingCheckInDate	ExtParm.Lodging	Conditional ⁷	Check-in date
LodgingCheckOutDate	ExtParm.Lodging	Conditional ⁷	Check-out date
LodgingRoomAmount	ExtParm.Lodging	Conditional ⁷	Nightly rate for one room
LodgingRoomTax	ExtParm.Lodging	Conditional ⁷	Reserved for future use
LodgingNoShowIndicator	ExtParm.Lodging	Conditional ⁷	No-show indicator
LodgingDuration	ExtParm.Lodging	Conditional ⁷	Number of hotel stay days (e.g. 03). Pass 01 for no-show.
LodgingCustomerName	ExtParm.Lodging	Conditional ⁷	Customer name
LodgingClientCode	ExtParm.Lodging	Conditional ⁷	Reserved for future use
LodgingExtraChargesDetail	ExtParm.Lodging	Conditional ⁷	Extra charges detail
LodgingExtraChargesAmounts	ExtParm.Lodging	Conditional ⁷	Reserved for future use
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Conditional ⁷	Prestigious property code
LodgingSpecialProgramCode	ExtParm.Lodging	Conditional ⁷	Special program code
LodgingChargeType	ExtParm.Lodging	Conditional ⁷	Charge type
TokenProvider	ExtParm.Token	Conditional ⁹	Token Provider enumeration Null = 0 ExpressPASS = 1 OmniToken = 2 Paymetric = 3 TransArmor = 4

Only one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or PaymentAccountID or EncryptedTrack1Data or EncryptedTrack2Data or EncryptedCardData.

² Required when Scheduled Task object is being included.

- 3 Used for Direct Marketing, MOTO, and E-Commerce transactions. Required when card number is manually keyed.
- 4 Used when desiring to associate a task to an already-existing ScheduledTaskGroupID.
- 5 Used when submitting a Healthcare transaction (TSYS, FDC, and Global platform only).
- 6 Used when submitting an Auto Rental transaction (TSYS and FDC platform only).
- 7 Used when submitting a Lodging transaction (TSYS and FDC platform only).
- 8 Lane number is required for some processors; including the lane number is never an error.
- 9 Used when submitting a TransArmor transaction (FDC platform only).

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code.
ExpressResponseMessage	Response	Returned	Express Response Message.
MostResponseCode	Response	Returned	Host Response Code.
* HostResponseMessage	Response	Returned	Host Response Message.
Express Transaction Date	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
MostBatchID HostBatchID	Response Batch	Returned	Unique batch identifier
MostItemID 1	Response.Batch	Returned	Unique batch item identifier
MostBatchAmount	Response.Batch	Returned	Total amount of items in batch
*AVSResponseCode	Response.Card	Conditional	Address Verification
		Requires Address Information to be returned	Service Response Code
CVVResponseCode	Response Card	Conditional	Card Verification Value Response Code
		Requires CVV input to be returned	response code
CAVVResponseCode	Response.Card	Conditional	Cardholder Authentication
		Requires CAVV input to be returned	Verification Value Response Code
© CardLogo	Response Card	Returned	Payment brand
* TransactionID	Response.Transaction	Returned	Unique transaction identifier
* ApprovalNumber	Response Transaction	Returned	Approval number assigned by card issuing bank.
ReferenceNumber	Response. Transaction	Conditional	Unique batch item
		Returns input value if supplied	identifier
AcquirerData	Response.Transaction	Returned	The Acquirer Reference Data returned here

Name	Class	Returned	Description
			must be captured by the POS device and input in subsequent settlement requests
ProcessorName	Response.Transaction	Returned	Name of processor
* TransactionStatus	Response Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction
ApprovedAmount	Response Transaction	Conditional	Approved amount
		May or may not be returned, depending on account	
CommercialCardResponseCode	Response.Transaction	Conditional	Commercial card
		TSYS or FDC or Vantiv only	response code
BalanceAmount	Response Transaction	Conditional	Balance amount
		May be returned with partial approval support	
BalanceCurrencyCode	Response. Transaction	Conditional	Balance currency code
		May be returned with partial approval support	
PaymentAccountID	Response PaymentAccount	Conditional	A unique system
		Returned if Scheduled Task class was supplied	generated identifier.
Danier and Danier and Danier	Response. Payment Account	Conditional	Payment Account
PaymentAccountReferenc eNumber		Returned if PaymentAccountID supplied	Reference Number
ScheduledTaskID	Response. Scheduled Task	Conditional	A unique system
		Returned if Scheduled Task class was supplied	generated identifier.
ScheduledTaskName	Response. Scheduled Task	Conditional	User generated name of
		Returned if passed within Scheduled	the task.

Name	Class	Returned	Description
		Task class	
ScheduledTaskGroupID	Response ScheduledTask	Conditional	Express platform
		Returned if passed within Scheduled Task class	generates this.
BillingAddress1	Response. Address	Conditional	The street address used
		Returns input value if supplied	for billing purposes
BillingZipcode	Response. Address	Conditional	The zip code used for
		Returns input value if supplied	billing purposes
* TokenID	Response. Token	Returned	TransArmor Token ID
[♥] TokenProvider	Response.Token	Conditional Returns input	Token Provider enumeration
		value if supplied	Null = 0
			ExpressPASS = 1 OmniToken = 2
			Paymetric = 3
			TransArmor = 4
№ TAProviderID	Response.Token	Returned	TransArmor Token Provider ID

CreditCardAuthorization

An Authorization transaction is used to verify funds when the total amount of the purchase is unknown.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
▼ <u>TransactionAmount</u>	<u>Transaction</u>	Required	Amount of transaction. Depending on the merchant setup, a \$0.00 account verification may be used when address verification is performed.
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
SalesTaxAmount	Transaction	Optional	Sales tax amount. For Level II, submit amount for tax or 0.00 for tax-exempt.
TipAmount	Transaction	Optional	Tip amount
ConvenienceFeeAmount	Transaction	Optional	Convenience fee amount (Vantiv only)
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Optional ²	Unique numeric value assigned by the transaction sender
CommercialCardCustomerCode	<u>Transaction</u>	Optional	Commercial Card Customer Code
PartialApprovedFlag	Transaction	Conditional	Indicates whether software supports partial approvals
MerchantVerificationValue	Transaction	Conditional ³	Healthcare Merchant Verification Value
MarketCode	Transaction	Required	Market Code
* TerminalID	Terminal	Required	Unique terminal identifier
<u>CardPresentCode</u>	<u>Terminal</u>	Required	Card Present Code
<u>CardholderPresentCode</u>	<u>Terminal</u>	Required	Cardholder Present Code
<u>CardInputCode</u>	<u>Terminal</u>	Required	Card Input Code
©PCVVPresenceCode	<u>Terminal</u>	Required	Card Verification Value Presence Code
<u>FTerminalCapabilityCode</u>	<u>Terminal</u>	Required	Terminal Capability Code

Name	Class	Required	Description
TerminalEnvironmentCode	Terminal	Required	Terminal Environment Code
<u>IPMotoECICode</u>	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
TerminalType	<u>Terminal</u>	Optional	Terminal Type
TerminalEncryptionFormat	Terminal	Optional	Not used
TerminalSerialNumber	Terminal	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	<u>Terminal</u>	Conditional ⁶	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Optional	Payment Account ID
Track1Data	Card	Conditional ¹	Track 1 data
	Card	Conditional ¹	Track 2 data
MagneprintData	Card	Conditional ¹	Magneprint data
CardNumber	Card	Conditional ¹	Account number on card
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
EncryptedTrack1Data	<u>Card</u>	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	<u>Card</u>	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	<u>Card</u>	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerialNumber	<u>Card</u>	Conditional ¹	Card Data Key Serial Number from integrated encryption device
EncryptedFormat	<u>Card</u>	Optional	Encryption format from integrated encryption device
	<u>Card</u>	Optional	Card Verification Value:
			Visa CVV2, MasterCard CVC2, Discover CID
Ø <u>CAVV</u>	<u>Card</u>	Optional	Card Authorization Verification Value
⋄ <u>XID</u>	<u>Card</u>	Optional	Verified by Visa transaction identifier
BillingName	Address	Optional	The name used for billing purposes. Recommend submitting for Level II processing.
BillingAddress1	Address	Conditional Needed for AVS	The street address used for billing purposes
BillingAddress2	<u>Address</u>	Optional	The street address used for billing purposes
BillingCity	Address	Optional	The city name used for billing purposes

Name	Class	Required	Description
BillingState	<u>Address</u>	Optional	The state name used for billing purposes
₱ BillingZipcode	Address	Conditional	The zip code used for billing purposes
		Needed for AVS	
BillingEmail	Address	Optional	The email address used for billing purposes
BillingPhone	<u>Address</u>	Optional	The phone number used for billing purposes
ShippingName	Address	Optional	The name used for billing purposes
ShippingAddress1	Address	Optional	The street address used for Shipping purposes
ShippingAddress2	Address	Optional	The street address used for Shipping purposes
ShippingCity	<u>Address</u>	Optional	The city name used for Shipping purposes
ShippingState	Address	Optional	The state name used for Shipping purposes
ShippingZipcode	<u>Address</u>	Optional	The zip code used for Shipping purposes. Recommend submitting for Level II processing.
ShippingEmail	<u>Address</u>	Optional	The email address used for Shipping purposes
ShippingPhone	<u>Address</u>	Optional	The phone number used for Shipping purposes
<u>IP</u> HealthcareFlag	ExtParm.Healthcare	Conditional ³	If this is set to True, the Healthcare parameters should be submitted.
<u>HealthcareFirstAccountType</u>	ExtParm.Healthcare	Conditional ³	Healthcare account type
HealthcareFirstAmountType	ExtParm.Healthcare	Conditional ³	Healthcare amount type
	ExtParm.Healthcare	Conditional ³	Currency Code. Use 840 for U.S.
<u>■ HealthcareFirstAmountSign</u>	ExtParm.Healthcare	Conditional ³	Positive or negative amount
HealthcareFirstAmount	ExtParm.Healthcare	Conditional ³	Healthcare amount
HealthcareSecondAccountType	ExtParm.Healthcare	Conditional ³	Healthcare account type
HealthcareSecondAmountType	ExtParm.Healthcare	Conditional ³	Healthcare amount type
HealthcareSecondCurrencyCode	ExtParm.Healthcare	Conditional ³	Currency Code. Use 840 for U.S.
HealthcareSecondAmountSign	ExtParm.Healthcare	Conditional ³	Positive or negative amount
HealthcareSecondAmount	ExtParm.Healthcare	Conditional ³	Healthcare amount
HealthcareThirdAccountType	ExtParm.Healthcare	Conditional ³	Healthcare account type
■ HealthcareThirdAmountType	ExtParm.Healthcare	Conditional ³	Healthcare amount type
HealthcareThirdCurrencyCode	ExtParm.Healthcare	Conditional ³	Currency Code. Use 840 for U.S.
HealthcareThirdAmountSign	ExtParm.Healthcare	Conditional ³	Positive or negative amount

Name	Class	Required	Description
HealthcareThirdAmount	ExtParm.Healthcare	Conditional ³	Healthcare amount
HealthcareFourthAccountType	ExtParm.Healthcare	Conditional ³	Healthcare account type
HealthcareFourthAmountType	ExtParm.Healthcare	Conditional ³	Healthcare amount type
HealthcareFourthCurrencyCode	ExtParm.Healthcare	Conditional ³	Currency Code. Use 840 for U.S.
HealthcareFourthAmountSign	ExtParm.Healthcare	Conditional ³	Positive or negative amount
HealthcareFourthAmount	ExtParm.Healthcare	Conditional ³	Healthcare amount
AutoRentalDuration	ExtParm.AutoRental	Conditional ⁴	Number of rental days (e.g. 03). Pass 01 for no-show rentals.
LodgingDuration	ExtParm.Lodging	Conditional ⁵	Number of days
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Conditional ⁵	Prestigious property code
TokenProvider	ExtParm.Token	Conditional ⁷	Token Provider enumeration Null = 0
			ExpressPASS = 1
			OmniToken = 2
			Paymetric = 3
			TransArmor = 4

- Only one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or PaymentAccountID or EncryptedTrack1Data or EncryptedTrack2Data or EncryptedCardData.
- ² Used for Direct Marketing, MOTO, and E-Commerce transactions. Required when card number is manually keyed.
- ³ Used when submitting a Healthcare transaction (TSYS, FDC, and Global platform only).
- Used when submitting an Auto Rental transaction (TSYS and FDC platform only).
- ⁵ Used when submitting a Lodging transaction (TSYS and FDC platform only).
- ⁶ Lane number is required for some processors; including the lane number is never an error.
- 7 Used when submitting a TransArmor transaction (FDC platform only).

Name	Class	Returned	Description
* ExpressResponseCode	Response	Returned	Express Response Code.
ExpressResponseMessage	Response	Returned	Express Response Message.
MostResponseCode MostResponseCode	Response	Returned	Host Response Code.
₱ HostResponseMessage	Response	Returned	Host Response Message.
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	<u>Response</u>	Returned	Express transaction UTC time zone
	Response.Card	Conditional Requires Address Information to be returned	Address Verification Service Response Code
	Response.Card	Conditional	Card Verification Value
CVVResponsecode		Requires CVV input to be returned	Response Code
	Response. Card	Conditional	Cardholder Authentication
		Requires CAVV input to be returned	Verification Value Response Code
CardLogo	Response.Card	Returned	Payment brand
TransactionID	Response Transaction	Returned	Unique transaction identifier
ApprovalNumber	Response.Transaction	Returned	Approval number assigned by card issuing bank.
ReferenceNumber	Response.Transaction	Conditional	Unique batch item identifier
		Returns input value if supplied	
PaymentAccountReferenceNumber	Response.PaymentAccount	Conditional	Payment Account Reference Number
		Returned if PaymentAccountID supplied	Number
AcquirerData	Response.Transaction	Returned	The Acquirer Reference Data returned here must be captured by the POS device and input in subsequent settlement requests
ProcessorName	Response Transaction	Returned	Name of processor
* TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
* TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction
ApprovedAmount	Response Transaction	Conditional	Approved amount
		May or may not be returned, depending on account	

Name	Class	Returned	Description
CommercialCardResponseCode	Response Transaction	Conditional	Commercial card response
		TSYS or FDC or Vantiv only	code
₱ BalanceAmount	Response Transaction	Conditional	Balance amount
		May be returned with partial approval support	
₱ BalanceCurrencyCode	Response.Transaction	Conditional	Balance currency code
		May be returned with partial approval support	
BillingAddress1	Response Address	Conditional	The street address used for
		Returns input value if supplied	billing purposes
₱ BillingZipcode	illingZipcode Response_Address Conditional		The zip code used for billing
		Returns input value if supplied	purposes
* TokenID	Response.Token	Returned	TransArmor Token ID
* TokenProvider	<u>Response.Token</u>	Conditional Returns input value if supplied	Token Provider enumeration Null = 0 ExpressPASS = 1 OmniToken = 2 Paymetric = 3 TransArmor = 4
TAProviderID	Response.Token	Returned	The zip code used for billing purposes

CreditCardAuthorizationCompletion

An *Authorization Completion* transaction is used to complete the final amount of a previously Authorized transaction.

Example: Restaurants use *Authorization Completions* to enter in the final total amount after a customer has added a tip and signed the receipt.

An Authorization Completion can be entered for any amount up to an allowable excess limit. The allowable excess limit is determined by the association.

If the final amount exceeds the *Authorization* allowable excess limit, then merchant can either enter a *Sale* or *Force* transaction for the full amount.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
* AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
№ ShiftID	Transaction	Optional	Shift ID
Sales Tax Amount	Transaction	Optional	Sales tax amount. For Level II, submit amount for tax or 0.00 for tax-exempt.
TipAmount	Transaction	Optional	Tip amount
© ConvenienceFeeAmount	Transaction	Optional	Convenience fee amount (Vantiv only)
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
* <u>TicketNumber</u>	<u>Transaction</u>	Optional ⁴	Unique numeric value assigned by the transaction sender
CommercialCardCustomerCode	Transaction	Optional	Commercial Card Customer Code
*TransactionID	<u>Transaction</u>	Required	TransactionID returned in original transaction.
■ MarketCode	Transaction	Required	Market Code
DuplicateOverrideFlag	Transaction	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
* TerminalID	<u>Terminal</u>	Required	Unique terminal identifier

Name	Class	Required	Description
CardPresentCode	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
CardInputCode	Terminal	Required	Card Input Code
<u> TerminalCapabilityCode</u>	<u>Terminal</u>	Required	Terminal Capability Code
TerminalEnvironmentCode	Terminal	Required	Terminal Environment Code
MotoECICode	Terminal	Required	Mail Order Telephone Order Electronic Indicator Code
<u>FTerminalType</u>	Terminal	Optional	Terminal Type
<u> LaneNumber</u>	<u>Terminal</u>	Conditional ⁵	Lane number
LodgingAgreementNumber	ExtParm.Lodging	Conditional ¹	Agreement (Folio) number
LodgingCheckInDate	ExtParm.Lodging	Conditional ¹	Check-in date
LodgingCheckOutDate	ExtParm.Lodging	Conditional ¹	Check-out date
LodgingRoomAmount	ExtParm.Lodging	Conditional ¹	Nightly rate for one room
LodgingRoomTax	ExtParm.Lodging	Conditional ¹	Reserved for future use
LodgingNoShowIndicator	ExtParm.Lodging	Conditional ¹	No-show indicator
LodgingDuration	ExtParm.Lodging	Conditional ¹	Number of days
LodgingCustomerName	ExtParm.Lodging	Conditional ¹	Customer name
LodgingClientCode	ExtParm.Lodging	Conditional ¹	Reserved for future use
LodgingExtraChargesDetail	ExtParm.Lodging	Conditional ¹	Extra charges detail
LodgingExtraChargesAmounts	ExtParm.Lodging	Conditional ¹	Reserved for future use
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Conditional ¹	Prestigious property code
LodgingSpecialProgramCode	ExtParm.Lodging	Conditional ¹	Special program code
LodgingChargeType	ExtParm.Lodging	Conditional ¹	Charge type
AutoRentalAgreementNumber	ExtParm.AutoRental	Conditional ²	Rental agreement number
AutoRentalNoShowIndicator	ExtParm.AutoRental	Conditional ²	No show indicator
AutoRentalExtraChargesDetail	ExtParm.AutoRental	Conditional ²	Extra charges detail
AutoRentalPickupDate	ExtParm.AutoRental	Conditional ²	Rental pickup date. Format is YYYYMMDD.
AutoRentalCustomerName	ExtParm.AutoRental	Conditional ²	Name of the person or business with the reservation
AutoRentalReturnCity	ExtParm.AutoRental	Conditional ²	City where vehicle was returned
<u>AutoRentalReturnState</u>	ExtParm.AutoRental	Conditional ²	State where vehicle was returned
AutoRentalReturnLocationID	ExtParm.AutoRental	Conditional ²	Rental location ID
AutoRentalDuration	ExtParm.AutoRental	Conditional ²	Number of rental days (e.g. 03). Pass 01 for no-show rentals.

Name	Class	Required	Description
AutoRentalPickupLocation	ExtParm.AutoRental	Conditional ²	Name of business where vehicle was picked up
AutoRentalPickupCity	ExtParm.AutoRental	Conditional ²	Name of city, town, or village where vehicle was picked up
AutoRentalPickupState	ExtParm.AutoRental	Conditional ²	Abbreviation of state, province, or other country subdivision where vehicle was picked up
AutoRentalPickupCountryCode	ExtParm.AutoRental	Conditional ²	Country code where vehicle was picked up. For US, use 840.
AutoRentalPickupTime	ExtParm.AutoRental	Conditional ²	Local time when vehicle was picked up.
AutoRentalReturnCountryCode	ExtParm.AutoRental	Conditional ²	Country code where vehicle was returned. For US, use 840.
AutoRentalReturnDate	ExtParm.AutoRental	Conditional ²	Date when vehicle was returned. Format YYYYMMDD.
AutoRentalReturnTime	ExtParm.AutoRental	Conditional ²	Local time when vehicle was returned
AutoRentalVehicleClassCode	ExtParm.AutoRental	Conditional ²	Classification of rental vehicle
AutoRentalDistance	ExtParm.AutoRental	Conditional ²	Distance traveled during rental period
AutoRentalDistanceUnit	ExtParm.AutoRental	Conditional ²	Unit of measurement applicable to distance traveled
₽ AutoRentalAuditAdjustmentCode	ExtParm.AutoRental	Conditional ²	Indicates if adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement
AutoRentalAuditAdjustmentAmount	ExtParm.AutoRental	Conditional ²	Adjustment amount for mileage, fuel, vehicle damage, etc. after a vehicle was returned
Level III EnhancedData	ExtParm EnhancedData	Conditional ³	See Level III format in Section 4.1 for EnhancedData Extended Parameters input fields. XML only via TSYS, FDC, or Vantiv.

¹ Used when submitting a Lodging transaction (TSYS and FDC platform only).

² Used when submitting an Auto Rental transaction (TSYS and FDC platform only)

 $^{^{3}}$ Used when submitting a Level III transaction (TSYS, FDC, or Vantiv platform only)

⁴ Used for Direct Marketing, MOTO, and E-Commerce transactions. Required when card number is manually keyed.

 $^{^{\}bf 5}$ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response Card	Returned	Payment brand
HostBatchID	Response Batch	Returned	Unique batch identifier
* HostItemID	Response Batch	Returned	Unique batch item identifier
HostBatchAmount	Response. Batch	Returned	Total amount of items in batch
TransactionID	Response Transaction	Returned	Unique transaction identifier
AcquirerData	Response.Transaction	Returned	The Acquirer Reference Data returned here must be captured by the POS device and input in subsequent settlement requests
ReferenceNumber	Response Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response Transaction	Returned	Name of processor
▼ TransactionStatus ■ TransactionStatus	Response. Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response Transaction	Returned	Status/State of transaction

CreditCardCredit

A *Credit* transaction is used to refund a cardholder for a previous transaction.

Example: A customer returns merchandise and wants a refund.

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
OriginalAuthorizedAmount	Transaction	Optional	Original Authorization Amount
SalesTaxAmount	Transaction	Optional	Sales tax amount
ConvenienceFeeAmount	<u>Transaction</u>	Optional	Convenience fee amount (Vantiv only)
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
ApprovalNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Optional ²	Unique numeric value assigned by the transaction sender
₽ MarketCode	Transaction	Required	Market Code
[€]DuplicateOverrideFlag	Transaction	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
▼ TerminalID	Terminal	Required	Unique terminal identifier
CardPresentCode	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
CardInputCode	Terminal	Required	Card Input Code
©CVVPresenceCode	<u>Terminal</u>	Required	Card Verification Value Presence Code
**TerminalCapabilityCode	<u>Terminal</u>	Required	Terminal Capability Code
<u>FTerminalEnvironmentCode</u>	Terminal	Required	Terminal Environment Code

Name	Class	Required	Description
₽ MotoECICode	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
TerminalType	Terminal	Optional	Terminal Type
<u>TerminalEncryptionFormat</u>	Terminal	Optional	Not used
TerminalSerialNumber	<u>Terminal</u>	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	Terminal	Conditional	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Optional ³	Payment Account ID
* Track1Data	Card	Conditional ¹	Track 1 data
₱ Track2Data	<u>Card</u>	Conditional ¹	Track 2 data
MagneprintData	Card	Conditional ¹	Magneprint data
CardNumber	Card	Conditional ¹	Account number on card
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
EncryptedTrack1Data	Card	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	Card	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	Card	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
Card Data Key Serial Number	Card	Conditional ¹	Card Data Key Serial Number from integrated encryption device
 <u>EncryptedFormat</u>	Card	Optional	Encryption format from integrated encryption device

¹ Only one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or PaymentAccountID or EncryptedTrack1Data or EncryptedCardData.

² Used for Direct Marketing, MOTO, and E-Commerce transactions. Required when card number is manually keyed.

³ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
₱ HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo CardLogo	Response Card	Returned	Payment brand
MostBatchID HostBatchID	Response.Batch	Returned	Unique batch identifier
MostItemID	Response.Batch	Returned	Unique batch item identifier
HostBatchAmount	Response.Batch	Returned	Batch Amount
TransactionID	Response.Transaction	Returned	Unique transaction identifier
ApprovalNumber	Response. Transaction	Returned	Approval number assigned by card issuing bank.
ReferenceNumber	Response. Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response. Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
PaymentAccountReferenceeNumber	Response Payment Account	Conditional Returned if PaymentAccountID supplied	Payment Account Reference Number
BillingAddress1	Response Address	Conditional Returns input value if supplied	The street address used for billing purposes
BillingZipcode	Response. Address	Conditional Returns input value if supplied	The zip code used for billing purposes

CreditCardReturn

A *CreditCardReturn* transaction is similar to a *CreditCardCredit* transaction in that it is used to refund a cardholder for a previous transaction. Unlike the *CreditCardCredit* transaction, the *CreditCardReturn* transaction requires a *TransactionID* for the previous dependent transaction. Note that the *CreditCardReturn* method can be used only for transactions that are no greater than 120 days old.

Example: A customer returns merchandise and wants a refund.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
SalesTaxAmount	Transaction	Optional	Sales tax amount
ConvenienceFeeAmount	Transaction	Optional	Convenience fee amount (Vantiv only)
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TransactionID	Transaction	Required	TransactionID returned in original transaction.
* <u>TicketNumber</u>	Transaction	Optional ¹	Unique numeric value assigned by the transaction sender
 MarketCode	Transaction	Required	Market Code
DuplicateOverrideFlag	Transaction	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
TerminalID	Terminal	Required	Unique terminal identifier
CardPresentCode	<u>Terminal</u>	Required	Card Present Code
<u>CardholderPresentCode</u>	<u>Terminal</u>	Required	Cardholder Present Code
 CardInputCode	Terminal	Required	Card Input Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
TerminalEnvironmentCode	Terminal	Required	Terminal Environment Code
<u>MotoECICode</u>	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code

Name	Class	Required	Description
<u> TerminalType</u>	Terminal	Optional	Terminal Type
<u> </u>	Terminal	Conditional ²	Lane number

¹ Used for Direct Marketing, MOTO, and E-Commerce transactions. Required when card number is manually keyed.

Name	Class	Returned	Description
ResponseCode	Response	Returned	Express Response Code.
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response Card	Returned	Payment brand
HostBatchID	Response Batch	Returned	Unique batch identifier
MostItemID HostItemID	Response Batch	Returned	Unique batch item identifier
HostBatchAmount	Response Batch	Returned	Batch amount
TransactionID	Response Transaction	Returned	Unique transaction identifier
ApprovalNumber	Response. Transaction	Returned	Approval number assigned by card issuing bank
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response Transaction	Returned	Status/State of transaction

 $^{^{\}mathbf{2}}$ Lane number is required for some processors; including the lane number is never an error.

CreditCardAdjustment

The *Adjustment* transaction is used to edit the information on the host for a previously completed credit card transaction (e.g. Level III Line Item Detail).

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
OriginalAuthorizedAmount	Transaction	Optional	Original Authorization Amount
SalesTaxAmount	Transaction	Optional	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TransactionID	Transaction	Required	TransactionID returned in original transaction.
TicketNumber	Transaction	Required	Ticket Number
© CommercialCardCustom erCode	Transaction	Required	Commercial Card Customer Code
<u>■ MarketCode</u>	Transaction	Required	Market Code
* TerminalID	Terminal	Required	Unique terminal identifier
CardPresentCode	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
₽ CardInputCode	Terminal	Required	Card Input Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
<u> TerminalEnvironmentCode</u>	Terminal	Required	Terminal Environment Code
■ PMotoECICode	Terminal	Required	Mail Order Telephone Order Electronic Indicator Code
<u> TerminalType</u>	Terminal	Optional	Terminal Type
LaneNumber	Terminal	Conditional ²	Lane number
Level III EnhancedData	ExtParm EnhancedData	Optional ¹	See Level III format in Section 4.1 for EnhancedData Extended Parameters input fields. XML only

Name	Class	Required	Description
			via TSYS, FDC, or Vantiv.

 $^{^{}f 1}$ Used when submitting a Level III transaction (TSYS, FDC, or Vantiv platforms only).

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
Express Transaction Timezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response Card	Returned	Payment brand
HostBatchID	Response. Batch	Returned	Unique batch identifier
HostItemID	Response Batch	Returned	Unique batch item identifier
TransactionID	Response Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response Transaction	Returned	Status/State of transaction

 $^{^{\}mathbf{2}}$ Lane number is required for some processors; including the lane number is never an error.

CreditCardVoid

A *Void* transaction is used to completely back out a previous *Sale* transaction. This applies to all transactions that were entered into the current batch, and only for the full dollar amount of the original transaction.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
▼ TransactionID ■ TransactionID	Transaction	Required	TransactionID returned in original transaction.
TerminalID	<u>Terminal</u>	Required	Unique terminal identifier
<u>CardPresentCode</u>	<u>Terminal</u>	Required	Card Present Code
CardholderPresentCode	<u>Terminal</u>	Required	Cardholder Present Code
<u>CardInputCode</u>	<u>Terminal</u>	Required	Card Input Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
<u>TerminalEnvironmentCode</u>	Terminal	Required	Terminal Environment Code
■ MotoECICode	Terminal	Required	Mail Order Telephone Order Electronic Indicator Code
<u> TerminalType</u>	<u>Terminal</u>	Optional	Terminal Type
LaneNumber	<u>Terminal</u>	Conditional ¹	Lane number

 $[\]mathbf{^{1}}\mathsf{Lane}$ number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response Card	Returned	Payment brand
₱ HostBatchID	Response.Batch	Returned	Unique batch identifier
MostItemID HostItemID	Response Batch	Returned	Unique batch item identifier
HostBatchAmount	Response Batch	Returned	Batch amount
TransactionID	Response Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
ApprovalNumber	Response Transaction	Returned	
AcquirerData	Response.Transaction	Returned	The Acquirer Reference Data returned here must be captured by the POS device and input in subsequent settlement requests
ProcessorName	Response Transaction	Returned	Name of processor
TransactionStatus	Response. Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response Transaction	Returned	Status/State of transaction

CreditCardAVSOnly

The Address Verification Service (AVS) Only transaction may be used by merchants to determine whether to accept a transaction. This type of transaction only returns AVS information and does not request authorization. The format of the AVS information is exactly the same as the format used for Sale transactions.

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	0.00 is the amount that should be sent in.
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Required	Ticket Number
<u>MarketCode</u>	Transaction	Required	Market Code
TerminalID	Terminal	Required	Unique terminal identifier
CardPresentCode	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
CardInputCode	Terminal	Required	Card Input Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
<u>TerminalEnvironmentCode</u>	Terminal	Required	Terminal Environment Code
■ MotoECICode	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
<u>TerminalEncryptionFormat</u>	Terminal	Optional	Not used
* TerminalSerialNumber	<u>Terminal</u>	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
<u> TerminalType</u>	Terminal	Optional	Terminal Type
₽ LaneNumber	<u>Terminal</u>	Conditional ²	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Optional	Payment Account ID
Track1Data	Card	Conditional ¹	Track 1 data

Name	Class	Required	Description
Track2Data	Card	Conditional ¹	Track 2 data
MagneprintData	Card	Conditional ¹	Magneprint data
CardNumber	Card	Conditional ¹	Account number on card
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
EncryptedTrack1Data	Card	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	Card	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	Card	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerialNumber	Card	Conditional ¹	Card Data Key Serial Number from integrated encryption device
[€] EncryptedFormat	Card	Optional	Encryption format from integrated encryption device
BillingName	Address	Optional	The name used for billing purposes
BillingAddress1	Address	Required	The street address used for billing purposes.
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	<u>Address</u>	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	<u>Address</u>	Required	The zip code used for billing purposes
BillingEmail	Address	Optional	The email address used for billing purposes
BillingPhone	Address	Optional	The phone number used for billing purposes
ShippingName	<u>Address</u>	Optional	The name used for billing purposes
ShippingAddress1	Address	Optional	The street address used for Shipping purposes
ShippingAddress2	<u>Address</u>	Optional	The street address used for Shipping purposes
ShippingCity	<u>Address</u>	Optional	The city name used for Shipping purposes
ShippingState	Address	Optional	The state name used for Shipping purposes

Name	Class	Required	Description
ShippingZipcode	Address	Optional	The zip code used for Shipping
			purposes

¹ Only one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear **or** Track1Data **or** Track2Data **or** MagneprintData **or** PaymentAccountID **or** EncryptedTrack1Data **or** EncryptedTrack2Data **or** EncryptedCardData.

Name	Class	Returned	Description
ResponseCode	Response	Returned	Express Response Code.
*ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
Express Transaction Date	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
AVSResponseCode	Response Card	Returned	
CardLogo	Response.Card	Returned	Payment brand
AcquirerData	Response.Transaction	Returned	The Acquirer Reference Data returned here must be captured by the POS device and input in subsequent settlement requests
ApprovalNumber	Response Transaction	Returned	
*TransactionID	Response.Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response Transaction	Returned	Name of processor
* TransactionStatus	Response. Transaction	Returned	Description of status/state of transaction
* TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
PaymentAccountReferenc	Response.PaymentAccount	Conditional Returned if	Payment Account Reference Number

² Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
<u>eNumber</u>		PaymentAccountID supplied	
BillingAddress1	Response. Address	Conditional Returns input value if supplied	The street address used for billing purposes
BillingZipcode	Response. Address	Conditional Returns input value if supplied	The zip code used for billing purposes

CreditCardReversal

There are three types of *Reversal* transactions:

- 1) Communications/System Reversals
- 2) Point of Sale (POS) Generated/Full Reversals
- 3) Partial Reversals
- A Communications/System Reversal is generated when there is a problem delivering the response back to the POS. It is the transaction requestor's responsibility to generate a *Reversal* transaction for all transactions that do not receive responses, or for those receiving an *ExpressResponseCode* of **1001** or **1002**.
- A Point of Sale (POS) Generated Reversal may be initiated to remove an approved transaction from a batch, even when the cardholder is not present. Reversals, however, must be done while the batch is open.
- A Partial reversal A Partial Reversal transaction decreases the total authorized amount of a
 previously authorized transaction. Partial reversals are required by some card associations to bring
 the authorized and clearing amounts into specific tolerances.

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
■ ReversalType	Transaction	Required	Type of reversal
ClerkNumber	Transaction	Optional	Clerk number
№ ShiftID	Transaction	Optional	Shift ID
▼ <u>TransactionAmount</u>	<u>Transaction</u>	Required	Original transaction amount if ReversalType= Full or System; New transaction amount if ReversalType=Partial

Name	Class	Required	Description
SalesTaxAmount	Transaction	Conditional Used when it was included in the original transaction	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Required	Ticket Number
* TransactionID	Transaction	Required when ReversalType=Partial or ReversalType=Full. Optional when ReversalType=System	TransactionID returned in original transaction.
■ MarketCode	Transaction	Required	Market Code
TerminalID	Terminal	Required	Unique terminal identifier
₽ CardPresentCode	<u>Terminal</u>	Required	Card Present Code
CardholderPresentCode	<u>Terminal</u>	Required	Cardholder Present Code
₽ CardInputCode	<u>Terminal</u>	Required	Card Input Code
<u>FICVVPresenceCode</u>	<u>Terminal</u>	Required	Card Verification Value Presence Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
TerminalEnvironme ntCode	<u>Terminal</u>	Required	Terminal Environment Code
 ■ MotoECICode	Terminal	Required	Mail Order Telephone Order Electronic Indicator Code
■ TerminalType	Terminal	Optional	Terminal Type
TerminalEncryption Format	Terminal	Optional	Not used
TerminalSerialNumber	<u>Terminal</u>	Conditional ³	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	Terminal	Conditional	Lane number
CardNumber	Card	Conditional ^{1 and 2}	Account number on card
ExpirationMonth	Card	Conditional ^{1 and 2}	Expiration month
ExpirationYear	Card	Conditional ^{1 and 2}	Expiration year
Track1Data	Card	Conditional ^{1 and 2}	Track 1 data
▼ Track2Data	Card	Conditional ^{1 and 2}	Track 2 data
MagneprintData	Card	Conditional ^{1 and 2}	Magneprint data
EncryptedTrack1Data	Card	Conditional ^{1 and 2}	Encrypted Track 1 Data from

Name	Class	Required	Description
			integrated encryption device
EncryptedTrack2Data	Card	Conditional ^{1 and 2}	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	Card	Conditional ^{1 and 2}	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerial Number	Card	Conditional ^{1 and 2}	Card Data Key Serial Number from integrated encryption device
EncryptedFormat	Card	Optional	Encryption format from integrated encryption device
PaymentAccountID	ExtParm.PaymentAccount	Conditional 1 and 2	Payment Account ID
<u> </u>	ExtParm.Token	Optional ⁴	Unique Token ID representing the card

Only one of the following field groups needs to be included: TransactionID or CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or PaymentAccountID or EncryptedTrack1Data or EncryptedTrack2Data or EncryptedCardData.

Name	Class	Returned	Description
ResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
© CardLogo	Response.Card	Returned	Payment brand
HostBatchID	Response.Batch	Returned	Unique batch identifier
₱ HostItemID	Response Batch	Returned	Unique batch item identifier
HostBatchAmount	Response.Batch	Returned	Batch amount
<u>TransactionID</u>	Response. Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response.Transaction	Conditional Returns input	User defined reference number

² Used only when ReversalType = System

³ Lane number is required for some processors; including the lane number is never an error.

⁴ Token ID is optional and can be passed only for reversals using TransArmor token by FDC TransArmor merchants.

Name	Class	Returned	Description
		value if supplied	
ApprovalNumber	Response Transaction	Returned	
AcquirerData	Response.Transaction	Returned	The Acquirer Reference Data returned here must be captured by the POS device and input in subsequent settlement requests
	Response Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction

Remarks

- By sending the previously authorized *TransactionID* in the optional field *TransactionID*, it is not necessary to store or send *CardNumber*, *Track1Data*, *Track2Data*, *ExpirationMonth*, *ExpirationYear*, or *AcquirerData*.
- Processing System Reversals:

CreditCardSale	If you did not receive a response from a <i>CreditCardSale</i> transaction, call the <i>CreditCardReversal</i> method with a ReversalType of "System".		
CreditCardCredit	If you did not receive a response from a call to one of		
CreditCardReturn	these method types, you should resend the transaction.		
CreditCardAuthorizationCompletion	If you get back an ExpressResponseCode of 22 or 23 (Duplicate AP or Duplicate) then you know the transaction went through successfully. Otherwise the response you get back will be the correct one for that transaction.		
	Note Make sure you are not using the duplicate override or duplicate disable option when you resend the transaction.		

CreditCardForce

Generally, *Force* transactions are used to insert previously authorized, but not captured, transactions into a batch. The following examples illustrate when to use a *Force* transaction:

- 1) **Timeout** If the merchant is unable to perform an online transaction.
- 2) **Referral** If the merchant receives an online referral message in response to an online transaction (i.e. Call ND, Call AE, etc).
- 3) Al Responses The Al response (Approved with positive ID) usually results from a person using a card for the first time.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ApprovalNumber ApprovalNumber	Transaction	Required	Original approval number. Provided during voice authorization. Capital letters and numbers only.
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
SalesTaxAmount	Transaction	Optional	Sales tax amount
ConvenienceFeeAmount	Transaction	Optional	Convenience fee amount (Vantiv only)
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
<u>TicketNumber</u>	Transaction	Required	Ticket Number
CommercialCardCustomerCode	Transaction	Optional	Commercial Card Customer Code
■ MarketCode	Transaction	Required	Market Code
■ DuplicateOverrideFlag	<u>Transaction</u>	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
* TerminalID	<u>Terminal</u>	Required	Unique terminal identifier
<u>CardPresentCode</u>	<u>Terminal</u>	Required	Card Present Code
 <u>CardholderPresentCode</u>	<u>Terminal</u>	Required	Cardholder Present Code

Name	Class	Required	Description
E CardInputCode	Terminal	Required	Card Input Code
<u>CVVPresenceCode</u>	<u>Terminal</u>	Required	Card Verification Value Presence Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
TerminalEnvironmentCode	Terminal	Required	Terminal Environment Code
<u>III MotoECICode</u>	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
TerminalType	Terminal	Optional	Terminal Type
TerminalEncryptionFormat	Terminal	Optional	Not used
TerminalSerialNumber	<u>Terminal</u>	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
<u>LaneNumber</u>	Terminal	Conditional ⁴	Lane number
▼ Track1Data	Card	Conditional ¹	Track 1 data
Track2Data	Card	Conditional ¹	Track 2 data
MagneprintData	Card	Conditional ¹	Magneprint data
CardNumber	Card	Conditional ¹	Account number on card
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
EncryptedTrack1Data	<u>Card</u>	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	Card	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	Card	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
© CardDataKeySerialNumber	Card	Conditional ¹	Card Data Key Serial Number from integrated encryption device
EncryptedFormat	Card	Optional	Encryption format from integrated encryption device
AutoRentalAgreementNumber	ExtParm.AutoRental	Conditional ²	Rental agreement number
AutoRentalNoShowIndicator	ExtParm.AutoRental	Conditional ²	No show indicator
AutoRentalExtraChargesDetail	ExtParm.AutoRental	Conditional ²	Extra charges detail
AutoRentalPickupDate	ExtParm.AutoRental	Conditional ²	Rental pickup date. Format is YYYYMMDD.
AutoRentalCustomerName	ExtParm.AutoRental	Conditional ²	Name of the person or business with the reservation
AutoRentalReturnCity	ExtParm.AutoRental	Conditional ²	City where vehicle was returned

Name	Class	Required	Description
AutoRentalReturnState	ExtParm.AutoRental	Conditional ²	State where vehicle was returned
AutoRentalReturnLocationID	ExtParm.AutoRental	Conditional ²	Rental location ID
AutoRentalDuration	ExtParm.AutoRental	Conditional ²	Number of rental days (e.g. 03). Pass 01 for no-show rentals.
AutoRentalPickupLocation	ExtParm.AutoRental	Conditional ²	Name of business where vehicle was picked up
AutoRentalPickupCity	ExtParm.AutoRental	Conditional ²	Name of city, town, or village where vehicle was picked up
AutoRentalPickupState	ExtParm.AutoRental	Conditional ²	Abbreviation of state, province, or other country subdivision where vehicle was picked up
AutoRentalPickupCountryCode	ExtParm.AutoRental	Conditional ²	Country code where vehicle was picked up. For US, use 840.
AutoRentalPickupTime	ExtParm.AutoRental	Conditional ²	Local time when vehicle was picked up.
AutoRentalReturnCountryCode	ExtParm.AutoRental	Conditional ²	Country code where vehicle was returned. For US, use 840.
AutoRentalReturnDate	ExtParm.AutoRental	Conditional ²	Date when vehicle was returned. Format YYYYMMDD.
AutoRentalReturnTime	ExtParm.AutoRental	Conditional ²	Local time when vehicle was returned
	ExtParm.AutoRental	Conditional ²	Classification of rental vehicle
<u>AutoRentalDistance</u>	ExtParm.AutoRental	Conditional ²	Distance traveled during rental period
AutoRentalDistanceUnit	ExtParm.AutoRental	Conditional ²	Unit of measurement applicable to distance traveled
<u>AutoRentalAuditAdjustmentCode</u>	ExtParm.AutoRental	Conditional ²	Indicates if adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement
AutoRentalAuditAdjustmentAmount	ExtParm.AutoRental	Conditional ²	Adjustment amount for mileage, fuel, vehicle damage, etc. after a vehicle was returned
LodgingAgreementNumber	ExtParm.Lodging	Conditional ³	Agreement (Folio) number
LodgingCheckInDate	ExtParm.Lodging	Conditional ³	Check-in date
LodgingCheckOutDate	ExtParm.Lodging	Conditional ³	Check-out date
LodgingRoomAmount	ExtParm.Lodging	Conditional ³	Nightly rate for one room
LodgingRoomTax	ExtParm.Lodging	Conditional ³	Reserved for future use
LodgingNoShowIndicator	ExtParm.Lodging	Conditional ³	No-show indicator
LodgingDuration	ExtParm.Lodging	Conditional ³	Number of days
LodgingCustomerName	ExtParm.Lodging	Conditional ³	Customer name
LodgingClientCode	ExtParm.Lodging	Conditional ³	Reserved for future use
LodgingExtraChargesDetail	ExtParm.Lodging	Conditional ³	Extra charges detail
LodgingExtraChargesAmounts	ExtParm.Lodging	Conditional ³	Reserved for future use
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Conditional ³	Prestigious property code

Name	Class	Required	Description
LodgingSpecialProgramCode	ExtParm.Lodging	Conditional ³	Special program code
LodgingChargeType	ExtParm.Lodging	Conditional ³	Charge type
Level III EnhancedData	ExtParm EnhancedData	Optional	See Level III format in Section 4.1 for EnhancedData Extended Parameters input fields. XML only via TSYS, FDC, or Vantiv.

Only one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or EncryptedTrack1Data or EncryptedTrack2Data or EncryptedCardData.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	<u>Response</u>	Returned	Express transaction UTC time zone
CardLogo	Response Card	Returned	Payment brand
HostBatchID	Response.Batch	Returned	Unique batch identifier
MostItemID HostItemID	Response Batch	Returned	Unique batch item identifier
TransactionID	Response Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response Transaction	Returned	Name of processor
*TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response Transaction	Returned	Status/State of transaction
BillingAddress1	Response.Address	Conditional Returns input value if supplied	The street address used for billing purposes
BillingZipcode	Response Address	Conditional	The zip code used for billing

² Used when submitting an Auto Rental transaction (TSYS and FDC platform only).

³ Used when submitting a Lodging transaction (TSYS and FDC platform only).

⁴ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
		Returns input value if supplied	purposes

CreditCardIncrementalAuthorization

- A *CreditCardIncrementalAuthorization* transaction is used to authorize an additional amount to an original **Auto Rental** or **Lodging** industry authorization transaction.
- This method is used on the **TSYS** or **FDC** platform only.
- Only Visa cards support incremental authorizations.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Additional (incremental) authorization amount requested
*TransactionID	Transaction	Required	TransactionID returned in original authorization.
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
SalesTaxAmount	Transaction	Optional	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Required	Ticket Number
■ MarketCode	Transaction	Required	Market Code
♦ TerminalID	Terminal	Required	Unique terminal identifier
<u> </u>	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
<u> </u>	Terminal	Required	Card Input Code
<u>EPCVVPresenceCode</u>	Terminal	Required	Card Verification Value Presence Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
■ TerminalEnvironmentCode	Terminal	Required	Terminal Environment Code
[₽] MotoECICode	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code

Name	Class	Required	Description
<u> TerminalType</u>	Terminal	Optional	Terminal Type
<u> </u>	Terminal	Conditional ³	Lane number
LodgingDuration	ExtParm.Lodging	Optional ¹	Number of additional days for the hotel stay
AutoRentalDuration	ExtParm.AutoRental	Optional ²	Number of additional days for the auto rental

 $^{^{\}mathbf{1}}\,$ Can be used when submitting a Lodging transaction (TSYS and FDC platform only).

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
HostTransactionDateTime	Response	Returned	Host transaction date and time
CardLogo	Response.Card	Returned	Payment brand
MostBatchID	Response Batch	Returned	Unique batch identifier
MostItemID HostItemID	Response.Batch	Returned	Unique batch item identifier
TransactionID	Response Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
ApprovalNumber	Response.Transaction	Returned	Approval number assigned by card issuing bank.
ProcessorName	Response Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
BillingAddress1	Response.Address	Conditional Returns input value if supplied	The street address used for billing purposes

² Can be used when submitting an Auto Rental transaction (TSYS and FDC platform only).

³ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
₱ BillingZipcode	Response. Address	Conditional Returns input value if supplied	The zip code used for billing purposes

CreditCardBalanceInquiry

A *CreditCardBalanceInquiry* transaction is used to obtain a balance amount on a **pre-paid debit card.** This method may not be available across all platforms.

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Transaction amount. Use 0.00
ClerkNumber	<u>Transaction</u>	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Optional	Unique numeric value assigned by the transaction sender
■ MarketCode	Transaction	Required	Market Code
TerminalID	Terminal	Required	Unique terminal identifier
CardPresentCode	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
CardInputCode	Terminal	Required	Card Input Code
■ PCVVPresenceCode	<u>Terminal</u>	Required	Card Verification Value Presence Code
TerminalCapabilityCode	Terminal	Required	Terminal Capability Code
TerminalEnvironme ntCode	Terminal	Required	Terminal Environment Code
<u>MotoECICode</u>	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
TerminalType	Terminal	Optional	Terminal Type
■ LaneNumber	Terminal	Conditional ²	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Conditional ¹	Payment Account ID

Name	Class	Required	Description
Track1Data	Card	Conditional ¹	Track 1 data
Track2Data	Card	Conditional ¹	Track 2 data
MagneprintData	Card	Conditional ¹	Magneprint data
CardNumber	Card	Conditional ¹	Account number on card
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
EncryptedTrack1Data	Card	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	Card	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	Card	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerialNumber	Card	Conditional ¹	Card Data Key Serial Number from integrated encryption device
EncryptedFormat	Card	Optional	Encryption format from integrated encryption device

Only one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or PaymentAccountID or EncryptedTrack1Data or EncryptedTrack2Data or EncryptedCardData.

² Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
HostTransactionDateTime	Response	Returned	Host transaction date and time
CardLogo	Response.Card	Returned	Payment brand
<u>HostBatchID</u>	Response Batch	Returned	Unique batch identifier
₱ HostItemID	Response.Batch	Returned	Unique batch item identifier
TransactionID	Response Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response. Transaction	Conditional Returns input value if supplied	User defined reference number
	Response Transaction	Returned	Approval number assigned by card issuing bank.
ApprovedAmount	Response. Transaction	Returned	Approved amount
ProcessorName	Response. Transaction	Returned	Name of processor
TransactionStatus	Response Transaction	Returned	Description of status/state of transaction
* TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction
BalanceAmount	Response. Transaction	Returned	Balance amount
BalanceCurrencyCode	Response. Transaction	Returned	Balance currency code

Debit

DebitCardSale

The *Sale* transaction is the most basic and most common transaction, used for the sale/purchase of goods or services. This type of transaction compares the cardholder's funds available to the amount specified in the transaction. If the amount specified is available, the transaction is approved by the cardholder's bank and an approval code is returned. If the amount is **not** available or the transaction is denied for any other reason by the bank, an error message containing the reason is returned.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
Sales Tax Amount	Transaction	Optional	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
[♠] TicketNumber	Transaction	Optional	Unique numeric value assigned by the transaction sender
■ MarketCode	Transaction	Required	Market Code
CashBackAmount	Transaction	Optional	Cash back amount
DuplicateOverrideFlag	Transaction	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
TerminalID	Terminal	Required	Unique terminal identifier
<u>CardPresentCode</u>	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
<u> CardInputCode</u>	Terminal	Required	Card Input Code
<u>CVVPresenceCode</u>	Terminal	Required	Card Verification Value Presence Code
<u>TerminalCapabilityCode</u>	Terminal	Required	Terminal Capability Code
<u>TerminalEnvironmentCode</u>	<u>Terminal</u>	Required	Terminal Environment Code
<u>MotoECICode</u>	Terminal	Required	Mail Order Telephone Order

Name	Class	Required	Description
			Electronic Indicator Code
TerminalType	Terminal	Optional	Terminal Type
TerminalEncryptionFormat	Terminal	Optional	Not used
TerminalSerialNumber	Terminal	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	<u>Terminal</u>	Conditional ¹	Lane number
PINBlock	Card	Required	Encrypted PIN Block
KeySerialNumber	Card	Required	Key Serial Number
▼ Track2Data	<u>Card</u>	Conditional Not needed if MagneprintData has been included	Track 2 data
MagneprintData	Card	Conditional Not needed if Track2Data has been included	Magneprint data
₱ EncryptedTrack2Data	Card	Conditional Not needed if Track2Data or MagneprintData has been included.	Encrypted Track 2 Data from integrated encryption device
CardDataKeySerialNumber	Card	Conditional Not needed if Track2Data or MagneprintData has been included.	Card Data Key Serial Number from integrated encryption device
EncryptedFormat	Card	Optional	Encryption format from integrated encryption device

 $^{^{\}mathbf{1}}$ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]

Name	Class	Returned	Description
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response.Card	Returned	Payment brand
HostBatchID	Response.Batch	Returned	Unique batch identifier
MostItemID	Response.Batch	Returned	Unique batch item identifier
HostBatchAmount	Response.Batch	Returned	Batch amount
TransactionID	Response Transaction	Returned	Unique transaction identifier
ApprovalNumber	Response.Transaction	Conditional Returns if assigned by bank	Approval number assigned by card issuing bank.
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response. Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction
BalanceAmount	Response.Transaction	Conditional May be returned with partial approval support	For future use
ApprovedAmount	Response.Transaction	Conditional May or may not be returned, depending on account	Approved amount
BalanceCurrencyCode	Response.Transaction	Conditional May be returned with partial approval support	For future use

■ DebitCardReturn

A *Return* transaction is used to refund a cardholder for a previous transaction. Debit card *Returns* must be done online. **Due to security reasons, this method may not be available on your merchant account.** Contact the Express Certification team for further details.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
OriginalAuthorizedAmount	Transaction	Optional	Original Authorization Amount
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
Sales Tax Amount	Transaction	Optional	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Optional	Unique numeric value assigned by the transaction sender
<u>MarketCode</u>	<u>Transaction</u>	Required	Market Code
DuplicateOverrideFlag	Transaction	Optional	Override a duplicate upon receiving a duplicate decline response
<u>DuplicateCheckDisableFlag</u>	<u>Transaction</u>	Optional	Bypass duplicate checking logic
TerminalID	<u>Terminal</u>	Required	Unique terminal identifier
<u> CardPresentCode</u>	Terminal	Required	Card Present Code
CardholderPresentCode	<u>Terminal</u>	Required	Cardholder Present Code
CardInputCode	<u>Terminal</u>	Required	Card Input Code
<u>EPCVVPresenceCode</u>	Terminal	Required	Card Verification Value Presence Code
<u> TerminalCapabilityCode</u>	Terminal	Required	Terminal Capability Code
<u> TerminalEnvironmentCode</u>	<u>Terminal</u>	Required	Terminal Environment Code
<u>III MotoECICode</u>	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
<u> TerminalType</u>	<u>Terminal</u>	Optional	Terminal Type
TerminalEncryptionFormat	<u>Terminal</u>	Optional	Not used

Name	Class	Required	Description
TerminalSerialNumber	Terminal	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	<u>Terminal</u>	Conditional ¹	Lane number
PINBlock	Card	Required	Encrypted PIN Block
KeySerialNumber	Card	Required	Key Serial Number
▼ Track2Data	<u>Card</u>	Conditional Not needed if MagneprintData has been included	Track 2 data
MagneprintData MagneprintData	<u>Card</u>	Conditional Not needed if Track2Data has been included	Magneprint data
EncryptedTrack2Data	<u>Card</u>	Conditional Not needed if Track2Data or MagneprintData has been included.	Encrypted Track 2 Data from integrated encryption device
CardDataKeySerialNumber	<u>Card</u>	Conditional Not needed if Track2Data or MagneprintData has been included.	Card Data Key Serial Number from integrated encryption device
■ EncryptedFormat	<u>Card</u>	Optional	Encryption format from integrated encryption device

 $^{^{\}mathbf{1}}\mathsf{Lane}$ number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response.Card	Returned	Payment brand
₱ HostBatchID	Response Batch	Returned	Unique batch identifier

Name	Class	Returned	Description
₱ HostItemID	Response Batch	Returned	Unique batch item identifier
HostBatchAmount	Response.Batch	Returned	
TransactionID	Response Transaction	Returned	Unique transaction identifier
* <u>ApprovalNumber</u>	Response. Transaction	Conditional Returns if assigned by bank	Approval number assigned by card issuing bank.
ReferenceNumber	Response Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response. Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction
ApprovedAmount	Response.Transaction	Conditional May or may not be returned, depending on account	Approved amount

DebitCardReversal

There are two types of Debit *Reversal* transactions:

- A Communication/System Reversal is generated when there is a problem delivering the response back to the POS. It is the transaction requestor's responsibility to generate a Reversal transaction for all transactions that do not receive responses, or for those receiving an *ExpressResponseCode* of **1001** or **1002**.
- A Point of Sale (POS) Generated/Full Reversal may be initiated to remove an approved transaction from a batch, even when the cardholder is not present. Reversals, however, must be done while the batch is open.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
₽ ReversalType	Transaction	Required	Type of reversal. In the case of Debit this is always set to ReversalType = System or ReversalType = Full
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
<u> </u>	Transaction	Conditional Used for Full Reversal only	Unique transaction identifier
Transaction Amount	Transaction	Required	Original transaction amount
SalesTaxAmount	Transaction	Conditional Used when it was included in the original transaction	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Optional	Unique numeric value assigned by the transaction sender
■ MarketCode	Transaction	Required	Market Code
<u>▼ TerminalID</u>	<u>Terminal</u>	Required	Unique terminal identifier
<u> </u>	<u>Terminal</u>	Required	Card Present Code
CardholderPresentCode	<u>Terminal</u>	Required	Cardholder Present Code

Name	Class	Required	Description
<u>CardInputCode</u>	Terminal	Required	Card Input Code
<u>TerminalCapabilityCode</u>	Terminal	Required	Terminal Capability Code
<u> TerminalEnvironmentCode</u>	Terminal	Required	Terminal Environment Code
₽ MotoECICode	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
**TerminalEncryptionFormat	Terminal	Optional	Not used
* TerminalSerialNumber	<u>Terminal</u>	Conditional Submit with encrypted data	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	Terminal	Conditional ¹	Lane number
© CardNumber	Card	Conditional Not needed if optional field TransactionID has been included.	Account number on card
ExpirationMonth	Card	Conditional Not needed if optional field MagneprintData or Track2Data has been included	Expiration month
ExpirationYear	Card	Conditional Not needed if optional field MagneprintData or Track2Data has been included	Expiration year
▼ Track2Data	Card	Conditional Not needed if MagneprintData has been included, OR if CardNumber, ExpirationMonth, and ExpirationYear have been included. Used only when ReversalType = System.	Track 2 data
MagneprintData	Card	Conditional Not needed if Track2Data has been included, OR if CardNumber, ExpirationMonth, and ExpirationYear	Magneprint data

Name	Class	Required	Description
		have been included. Used only when ReversalType = System.	
	Card	Conditional Not needed if Track2Data has been included, OR if CardNumber, ExpirationMonth, and ExpirationYear have been included. Used only when ReversalType = System.	Magneprint data
EncryptedTrack2Data	<u>Card</u>	Conditional	Encrypted Track 2 Data from integrated encryption device
CardDataKeySerialNumber	Card	Conditional	Card Data Key Serial Number from integrated encryption device
■ EncryptedFormat	<u>Card</u>	Optional	Encryption format from integrated encryption device
▼ TransactionID ■ TransactionID	Transaction	Conditional Used for Full Reversal only	Unique transaction identifier DDD

¹Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response Card	Returned	Payment brand
MostBatchID HostBatchID	Response.Batch	Returned	Unique batch identifier
MostItemID HostItemID	Response.Batch	Returned	Unique batch item identifier
TransactionID	Response. Transaction	Returned	Unique transaction identifier

Name	Class	Returned	Description
ReferenceNumber	Response Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response. Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response Transaction	Returned	Status/State of transaction

Remarks

Processing Reversals:

DebitCardSale	If you did not receive a response from a <i>DebitCardSale</i> transaction, call the <i>DebitCardReversal</i> method with a <i>ReversalType</i> of "System".
DebitCardReturn	If you did not receive a response from a <i>DebitCardReturn</i> method call, you should just resend the transaction. If you get back an <i>ExpressResponseCode</i> of 22 or 23 (Duplicate AP or Duplicate) then you know the Transaction went through successfully. Otherwise the response you get back will be the correct one for that transaction. (Note: make sure you are not using the duplicate override option or the duplicate disable option when you resend the transaction.)

DebitCardPinlessSale

- The *DebitCardPinlessSale* transaction allows a debit card transaction to be processed without a PIN in a card-not-present environment (eCommerce or Direct Marketing/Phone/IVR transaction).
- This method is available only to specific merchant industry types, and only on specific debit card networks (such as NYCE, PULSE, INTERLINK, and STAR).
- This method is available only on the Vantiv platform at this time. Please contact your Element/Vantiv sales rep for additional information.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
	Transaction	Optional	Shift ID
SalesTaxAmount	Transaction	Optional	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
* <u>TicketNumber</u>	Transaction	Optional	Unique numeric value assigned by the transaction sender
■ PMarketCode	Transaction	Required	Market Code
CashBackAmount	Transaction	Optional	Cash back amount
[┏] DuplicateOverrideFlag	Transaction	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
BillPayerAccountNumber	Transaction	Required	Bill payer account number
▼ TerminalID	<u>Terminal</u>	Required	Unique terminal identifier
■ PCardPresentCode	Terminal	Required	Card Present Code
CardholderPresentCode	Terminal	Required	Cardholder Present Code
■ CardInputCode	<u>Terminal</u>	Required	Card Input Code
<u>EP</u> CVVPresenceCode	Terminal	Required	Card Verification Value Presence Code
■ TerminalCapabilityCode	<u>Terminal</u>	Required	Terminal Capability Code
TerminalEnvironmentCode	<u>Terminal</u>	Required	Terminal Environment Code

Name	Class	Required	Description
₽ MotoECICode	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
TerminalType	<u>Terminal</u>	Optional	Terminal Type
TerminalEncryptionFormat	Terminal	Optional	Not used
*TerminalSerialNumber	Terminal	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	Terminal	Conditional ¹	Lane number
© CardNumber	Card	Conditional Some form of card entry is required	Account number on card
Encrypted Card Data	Card	Conditional Some form of card entry is required	Encrypted card data
ExpirationMonth	Card	Conditional	Expiration month
ExpirationYear	Card	Conditional	Expiration year
CardDataKeySerialNumber	Card	Conditional Used if Encrypted Card Data is included.	Card Data Key Serial Number from integrated encryption device
PaymentAccountID	ExtParm .PaymentAccount	Conditional Some form of card entry is required	Payment Account ID
[■]EncryptedFormat	<u>Card</u>	Conditional Used if Encrypted Card Data is included.	Encryption format from integrated encryption device

 $^{^{\}mathbf{1}} \mathsf{Lane}$ number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
₱ HostResponseCode	Response	Returned	Host Response Code
<u> </u>	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
	Response.Card	Returned	Payment brand
* TransactionID	Response. Transaction	Returned	Unique transaction identifier
* <u>ApprovalNumber</u>	Response. Transaction	Conditional Returns if assigned by bank	Approval number assigned by card issuing bank.
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
	Response Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
* TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
ApprovedAmount	Response.Transaction	Conditional May or may not be returned, depending on account	Approved amount
BillPayerAccountNumber	Response.Transaction	Returned	Bill payer account number
BalanceAmount	Response Transaction	Conditional May be returned with partial approval support	For future use
BalanceCurrencyCode	Response.Transaction	Conditional May be returned with partial approval support	For future use

DebitCardPinlessReturn

- The *DebitCardPinlessReturn* transaction allows a debit card transaction to be refunded without a PIN in a card-not-present environment (eCommerce or Direct Marketing/Phone/IVR transaction).
- This method is available only to specific merchant industry types, and only on specific debit card networks as with the *DebitCardPinlessSale* method (though note that not all debit networks support PINless refunds).
- This method is available only on the Vantiv platform at this time. Please contact your Element/Vantiv sales rep for additional information.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
Transaction Amount	Transaction	Required	Amount of transaction
ClerkNumber	<u>Transaction</u>	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
SalesTaxAmount	Transaction	Optional	Sales tax amount
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
TicketNumber	Transaction	Optional	Unique numeric value assigned by the transaction sender
■ MarketCode	Transaction	Required	Market Code
CashBackAmount	<u>Transaction</u>	Optional	Cash back amount
DuplicateOverrideFlag	Transaction	Optional	Override a duplicate upon receiving a duplicate decline response
DuplicateCheckDisableFlag	Transaction	Optional	Bypass duplicate checking logic
BillPayerAccountNumber	Transaction	Required	Bill payer account number
TerminalID	<u>Terminal</u>	Required	Unique terminal identifier
<u> </u>	Terminal	Required	Card Present Code
<u>CardholderPresentCode</u>	Terminal	Required	Cardholder Present Code
 <u>CardInputCode</u>	<u>Terminal</u>	Required	Card Input Code
<u>CVVPresenceCode</u>	<u>Terminal</u>	Required	Card Verification Value Presence Code
<u> TerminalCapabilityCode</u>	Terminal	Required	Terminal Capability Code

Name	Class	Required	Description
<u>TerminalEnvironmentCode</u>	Terminal	Required	Terminal Environment Code
■ PMotoECICode	<u>Terminal</u>	Required	Mail Order Telephone Order Electronic Indicator Code
TerminalType	Terminal	Optional	Terminal Type
TerminalEncryptionFormat	Terminal	Optional	Not used
TerminalSerialNumber	<u>Terminal</u>	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	Terminal	Conditional ¹	Lane number
	Card	Conditional Some form of card entry is required	Account number on card
EncryptedCardData	Card	Conditional Some form of card entry is required	Encrypted card data
ExpirationMonth	Card	Conditional	Expiration month
ExpirationYear	Card	Conditional	Expiration year
CardDataKeySerialNumber	<u>Card</u>	Conditional Not needed if CardNumber has been included.	Card Data Key Serial Number from integrated encryption device
PaymentAccountID	ExtParm .PaymentAccount	Conditional Some form of card entry is required	Payment Account ID
■ PEncryptedFormat	<u>Card</u>	Conditional Used if Encrypted Card Data is included.	Encryption format from integrated encryption device

¹ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]

Name	Class	Returned	Description
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
CardLogo	Response Card	Returned	Payment brand
TransactionID	Response. Transaction	Returned	Unique transaction identifier
ApprovalNumber	Response.Transaction	Conditional Returns if assigned by bank	Approval number assigned by card issuing bank.
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response. Transaction	Returned	Status/State of transaction
ApprovedAmount	Response.Transaction	Conditional May or may not be returned, depending on account	Approved amount
BillPayerAccountNumber	Response Transaction	Returned	Bill payer account number
₱ BalanceAmount	Response Transaction	Conditional May be returned with partial approval support	For future use
BalanceCurrencyCode	Response.Transaction	Conditional May be returned with partial approval support	For future use

Check

CheckVerification

A *Verification* transaction is used to verify the consumers account information. It **does not** move funds or originate a transaction.

Name	Class	Required	Example
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
ReferenceNumber	<u>Transaction</u>	Required	Unique value assigned by the transaction sender
■ MarketCode	Transaction	Required	Market Code
TerminalID	Terminal	Required	Unique terminal identifier
₽ ConsentCode	Terminal	Optional	Consumer consent code
<u> </u>	Terminal	Conditional ¹	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Optional	Payment Account ID
AccountNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Account Number of the consumers bank account.
RoutingNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Routing Number of the consumers bank.
CheckNumber	DemandDepositAccount	Optional	Check Number
<u>F</u> CheckType	<u>DemandDepositAccount</u>	Optional	Check Type (Personal or Business)
₽ DDAAccountType	<u>DemandDepositAccount</u>	Optional	DDAAccountType (Checking, Savings, etc)
BillingName	<u>Address</u>	Conditional	The name used for billing purposes. Format 'First Name Last Name'

Name	Class	Required	Example
BillingAddress1	Address	Optional	The street address used for billing purposes
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	Address	Optional	The city name used for billing purposes
BillingState	<u>Address</u>	Optional	The state name used for billing purposes
BillingZipcode	<u>Address</u>	Optional	The zip code used for billing purposes
BillingEmail	<u>Address</u>	Optional	The email address used for billing purposes
BillingPhone	Address	Optional	The phone number used for billing purposes. Format 'XXXYYYZZZZ'.
BirthDate	Identification	Optional	Birth Date. Format 'YYYYMMDD'.
DriversLicenseNumber	Identification	Optional	Drivers License Number
DriversLicenseState	Identification	Optional	2 digit state abbreviation.
▼ TaxIDNumber	<u>Identification</u>	Optional	SSN or Tax ID Number. No dashes.

¹Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	<u>Response</u>	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	<u>Response</u>	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
₱ HostBatchID	Response Batch	Conditional	Unique batch identifier
MostItemID	Response.Batch	Conditional	Unique batch item identifier
MostTransactionID	Response.Transaction	Conditional	Transaction identifier from the host processor

Name	Class	Returned	Description
* TransactionID	Response.Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response. Transaction	Conditional Returns input value if supplied	Unique batch item identifier
ProcessorName	Response. Transaction	Returned	Name of processor
▼ TransactionStatus ■ TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
PaymentAccountReferenceeNumber	Response.PaymentAccount	Conditional Returned if PaymentAccountID supplied	Payment Account Reference Number
BillingAddress1	Response Address	Conditional Returns input value if supplied	The street address used for billing purposes
BillingZipcode	Response Address	Conditional Returns input value if supplied	The zip code used for billing purposes

CheckSale

The *Sale* transaction is the most basic and most common transaction, used for the sale/purchase of goods or services.

Name	Class	Required	Example
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
₱ ShiftID	Transaction	Optional	Shift ID
ReferenceNumber	Transaction	Required	Unique value assigned by the transaction sender

Name	Class	Required	Example
■ MarketCode	Transaction	Required	Market Code
▼ TerminalID	<u>Terminal</u>	Required	Unique terminal identifier
₽ ConsentCode	<u>Terminal</u>	Optional	Consumer consent code
■ LaneNumber	<u>Terminal</u>	Conditional ³	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Optional	Payment Account ID
AccountNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Account Number of the consumer bank account.
RoutingNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Routing Number of the consumer bank.
CheckNumber	DemandDepositAccount	Optional	Check Number
€ PCheckType	<u>DemandDepositAccount</u>	Conditional Required if CheckNumber is included	Check Type (Personal or Business)
₽ DDAAccountType	<u>DemandDepositAccount</u>	Required	DDAAccountType (Checking, Savings, etc)
BillingName	Address	Conditional	The name used for billing purposes. Format 'First Name Last Name'
BillingAddress1	Address	Optional	The street address used for billing purposes
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	<u>Address</u>	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	<u>Address</u>	Optional	The zip code used for billing purposes
BillingEmail	<u>Address</u>	Optional	The email address used for billing purposes
BillingPhone	<u>Address</u>	Optional	The phone number used for billing purposes. Format 'XXXYYYZZZZ'.
BirthDate	<u>Identification</u>	Optional	Birth Date. Format 'YYYYMMDD'.
DriversLicenseNumber	Identification	Optional	Drivers License Number
DriversLicenseState	<u>Identification</u>	Optional	2 digit state abbreviation.

Name	Class	Required	Example
▼ TaxIDNumber	<u>Identification</u>	Optional	SSN or Tax ID Number. No dashes.
ScheduledTaskReferenceeNumber	ExtParm. Scheduled Task	Optional ²	User defined unique reference number that identifies this scheduled task.
RunFrequency	ExtParm.ScheduledTask	Optional ²	The frequency you want to task to run.
RunStartDate	ExtParm.ScheduledTask	Optional ²	The date you want the task to start.
RunUntilCancelFlag	ExtParm.ScheduledTask	Optional ²	A flag stating if this task should keep running at the selected Run Frequency until the user cancel it, or to just run for the given amount of RunCycles.
RunCycles	ExtParm.ScheduledTask	Optional ²	The amount of times you want the task to run.
[■] ScheduledTaskStatus	ExtParm.ScheduledTask	Optional ²	Active or Disabled

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
HostTransactionID	Response Transaction	Conditional	Transaction identifier from the host processor
TransactionID	Response Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response.Transaction	Conditional Returns input value if supplied	Unique batch item identifier

¹ Not Used
² Required when Scheduled Task object is being included.
³ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ProcessorName	Response Transaction	Returned	Name of processor
* TransactionStatus	Response Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
PaymentAccountID	Response.PaymentAccount	Conditional Returned if Scheduled Task class was supplied	A unique system generated identifier.
PaymentAccountReferenceeNumber	Response.PaymentAccount	Conditional Returned if PaymentAccountID supplied	Payment Account Reference Number
ScheduledTaskID	Response.ScheduledTask	Conditional Returned if Scheduled Task class was supplied	A unique system generated identifier.
BillingAddress1	Response Address	Conditional Returns input value if supplied	The street address used for billing purposes
BillingZipcode	Response. Address	Conditional Returns input value if supplied	The zip code used for billing purposes

CheckCredit

A *CheckCredit* transaction is used to refund a consumer for a previous transaction. A *Return* transaction is like a *Credit* transaction except it requires a *TransactionID* from the original dependent transaction.

Name	Class	Required	Example
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionAmount	Transaction	Required	Amount of transaction
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
ReferenceNumber	Transaction	Required	Unique value assigned by the transaction sender
MarketCode	Transaction	Required	Market Code
TerminalID	Terminal	Required	Unique terminal identifier
ConsentCode	Terminal	Optional	Consumer consent code
LaneNumber	Terminal	Conditional ¹	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Optional	Payment Account ID
AccountNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Account Number of the consumers bank account.
RoutingNumber	<u>DemandDepositAccount</u>	Conditional Not needed if PaymentAccountID has been included	Routing Number of the consumers bank.
CheckNumber	<u>DemandDepositAccount</u>	Optional	Check Number
 CheckType	<u>DemandDepositAccount</u>	Conditional Required if CheckNumber is included	Check Type (Personal or Business)
<u>DDAAccountType</u>	<u>DemandDepositAccount</u>	Required	DDAAccountType (Checking, Savings, etc)
BillingName	<u>Address</u>	Conditional	The name used for billing purposes. Format 'First Name Last Name'

Name	Class	Required	Example
BillingAddress1	Address	Optional	The street address used for billing purposes
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	Address	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	<u>Address</u>	Optional	The zip code used for billing purposes
BillingEmail	Address	Optional	The email address used for billing purposes
BillingPhone	Address	Optional	The phone number used for billing purposes. Format 'XXXYYYZZZZ'.
BirthDate	Identification	Optional	Birth Date. Format 'YYYYMMDD'.
DriversLicenseNumber	Identification	Optional	Drivers License Number
DriversLicenseState	Identification	Optional	2 digit state abbreviation.
* <u>TaxIDNumber</u>	<u>Identification</u>	Optional	SSN or Tax ID Number. No dashes.

¹Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	<u>Response</u>	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	<u>Response</u>	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
HostTransactionID	Response.Transaction	Conditional	Transaction identifier from the host processor
▼ TransactionID	Response.Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response.Transaction	Conditional Returns input	Unique batch item identifier

Name	Class	Returned	Description
		value if supplied	
ProcessorName	Response. Transaction	Returned	Name of processor
▼ TransactionStatus ■ TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction
PaymentAccountReferenceeNumber	Response.PaymentAccount	Conditional Returned if PaymentAccountID supplied	Payment Account Reference Number
BillingAddress1	Response.Address	Conditional Returns input value if supplied	The street address used for billing purposes
₱ BillingZipcode	Response.Address	Conditional Returns input value if supplied	The zip code used for billing purposes

CheckReturn

A *CheckReturn* transaction is used to refund a consumer for a previous transaction. A *Return* transaction requires a *TransactionID* from the original dependent transaction.

Name	Class	Required	Example
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TerminalID	Terminal	Required	Unique terminal identifier
₽ ConsentCode	Terminal	Optional	Consumer consent code
₽ LaneNumber	Terminal	Conditional ¹	Lane number
* TransactionID	Transaction	Required	TransactionID of the original sale transaction
▼ TransactionAmount ◆ → TransactionAmount → Transact	<u>Transaction</u>	Required	Amount of transaction (must match original transaction amount at this time)
ReferenceNumber	Transaction	Required	Unique value assigned by the transaction sender
ClerkNumber	Transaction	Optional	Clerk number

Name	Class	Required	Example
ShiftID	Transaction	Optional	Shift ID
■ MarketCode	Transaction	Required	Market Code

 $^{^{\}mathbf{1}}$ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	<u>Response</u>	Returned	Express transaction UTC time zone
HostTransactionID	Response Transaction	Conditional	Transaction identifier from the host processor
TransactionID	Response. Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response. Transaction	Returned	Unique value assigned by the transaction sender
ProcessorName	Response. Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction

CheckVoid

A *CheckVoid* transaction is used to completely back out a previous Check *Sale, Credit, or Return* transaction. This applies to all transactions that were entered into the current batch, and only for the full dollar amount of the original transaction.

Input Fields

Name	Class	Required	Example
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
TerminalID	Terminal	Required	Unique terminal identifier
LaneNumber	Terminal	Conditional ¹	Lane number
▼ TransactionID ■ TransactionID	Transaction	Required	TransactionID of the original sale, credit, or return transaction
ReferenceNumber	<u>Transaction</u>	Required	Unique value assigned by the transaction sender
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	<u>Transaction</u>	Optional	Shift ID

¹ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	<u>Response</u>	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
MostTransactionID	Response.Transaction	Conditional	Transaction identifier from the host processor
TransactionID	Response Transaction	Returned	Unique transaction

Name	Class	Returned	Description
			identifier
ReferenceNumber	Response.Transaction	Returned	Unique value assigned by the transaction sender
ProcessorName	Response. Transaction	Returned	Name of processor
*TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
* TransactionStatusCode	Response.Transaction	Returned	Status/State of transaction

CheckReversal

There is only one type of *Reversal* available for Check transactions and that is a **Communications/ System Reversal.**

A *Communications/System Reversal* is generated when there is a problem delivering the response back to the POS. It is the transaction requestor's responsibility to generate a *Reversal* transaction for all transactions that do not receive responses.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
₽ ReversalType	Transaction	Required	Type of reversal. In the case of Check, this value is always set to ReversalType = System
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
TransactionAmount	Transaction	Required	Original transaction amount
Sales Tax Amount	Transaction	Conditional Used when it was included in the original transaction	Sales tax amount
ReferenceNumber	Transaction	Required	Unique value assigned by the transaction sender
TransactionID	<u>Transaction</u>	Optional	TransactionID returned in original transaction
TerminalID	Terminal	Required	Unique terminal identifier

Name	Class	Required	Description
LaneNumber	Terminal	Conditional ¹	Lane number
PaymentAccountID	ExtParm.PaymentAccount	Optional	Payment Account ID

 $^{^{\}mathbf{1}}$ Lane number is required for some processors; including the lane number is never an error.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
TransactionID	Response Transaction	Returned	Unique transaction identifier
ReferenceNumber	Response. Transaction	Conditional Returns input value if supplied	User defined reference number
ProcessorName	Response Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response Transaction	Returned	Status/State of transaction

Reporting

TransactionQuery

The *TransactionQuery* method is used to retrieve transaction information based on Parameters object fields passed in. The *ReportingData* field will return transaction information in XML format, with a maximum of 1000 records returned.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
* AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
* AcceptorID	Credentials	Required	MerchantID
TransactionDateTimeBegin	<u>Parameters</u>	Conditional	Begin date/time of transaction range formatted [yyyy-MM-dd HH:mm:ss.fff]. Required if TransactionID or TransactionSetupID is not included.
* TransactionDateTimeEnd	<u>Parameters</u>	Conditional	End date/time of transaction range formatted [yyyy-MM-dd HH:mm:ss.fff]. Required if TransactionID or TransactionSetupID is not included.
* TransactionID	<u>Parameters</u>	Optional	Unique transaction identifier. If submitted, all other query fields are ignored.
▼ TerminalID	<u>Parameters</u>	Optional	Unique terminal identifier
ApplicationID	<u>Parameters</u>	Optional	Unique application identifier
ApprovalNumber	<u>Parameters</u>	Optional	Issuer assigned approval number
ApprovedAmount	<u>Parameters</u>	Optional	Approved transaction amount
ExpressTransactionDate	<u>Parameters</u>	Optional	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	<u>Parameters</u>	Optional	Express transaction time formatted [HHMMSS]
MostBatchID HostBatchID	<u>Parameters</u>	Optional	Unique host batch identifier
MostItemID HostItemID	Parameters	Optional	Unique host batch item identifier

Name	Class	Required	Description
ReferenceNumber	<u>Parameters</u>	Optional	User defined reference number
ShiftID	<u>Parameters</u>	Optional	Shift identifier
▼ TrackingID	<u>Parameters</u>	Optional	Internal transaction tracking identifier
Transaction Amount	<u>Parameters</u>	Optional	Dollar amount of transaction
**TransactionSetupID	<u>Parameters</u>	Optional	Transaction Setup ID (while only one successful transaction will be included, there could be multiple decline/error transactions, so the ExpressResponseCode, TransactionType, and TransactionStatus should be interrogated). If submitted, all other query fields are ignored.
₱ TransactionStatusCode	<u>Parameters</u>	Optional	Numeric-only status code of transaction
TransactionType	<u>Parameters</u>	Optional	Type of transaction
ReverseOrder	<u>Parameters</u>	Optional	Flag to query records in descending order

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Reporting Response Code
ExpressResponseMessage	Response	Returned	Reporting Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
Reporting Data	Response	Returned	Transaction information in XML format. A maximum of 1000 records can be returned.
ReportingID	Response	Returned	Unique reporting identifier

Services

PaymentAccountCreate

The PaymentAccountCreate method is used to create a new Payment Account record.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
<u>AccountID</u>	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
* AcceptorID	Credentials	Required	MerchantID
PaymentAccountType	PaymentAccount	Required	Payment Account Type
PaymentAccountReferenceNumber	<u>PaymentAccount</u>	Required	Payment Account Reference Number
₽ PASSUpdaterBatchStatus	<u>PaymentAccount</u>	Optional ²	Specifies whether or not the PASS record will be included in the next PASS Updater batch
■ PASSUpdaterOption	<u>PaymentAccount</u>	Optional ²	Specifies whether or not the PASS record will be updated automatically
CardNumber	Card	Conditional ¹	Card Number
ExpirationMonth	Card	Conditional ¹	Expiration Month (MM)
ExpirationYear	Card	Conditional ¹	Expiration Year (YY)
Track1Data	Card	Conditional ¹	Track1 Data
₱ Track2Data	Card	Conditional ¹	Track2 Data
MagneprintData	Card	Conditional ¹	Magneprint Data
EncryptedTrack1Data	Card	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	Card	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	<u>Card</u>	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerialNumber	Card	Conditional ¹	Card Data Key Serial

Name	Class	Required	Description
			Number from integrated encryption device
^{■P}EncryptedFormat	Card	Optional	Encryption format from integrated encryption device
* AccountNumber	<u>DemandDepositAccount</u>	Conditional - (Required if PaymentAccountType is a DDA account)	Account Number
RoutingNumber	DemandDepositAccount	Conditional - (Required if PaymentAccountType is a DDA account)	Routing Number
BillingName	Address	Optional	The name used for billing purposes
BillingAddress1	Address	Optional	The street address used for billing purposes
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	Address	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	Address	Optional	The zip code used for billing purposes
BillingEmail	Address	Optional	The email address used for billing purposes
BillingPhone	Address	Optional	The phone number used for billing purposes
ShippingName	Address	Optional	The name used for shipping purposes
ShippingAddress1	Address	Optional	The street address used for Shipping purposes
ShippingAddress2	Address	Optional	The street address used for Shipping purposes
ShippingCity	Address	Optional	The city name used for Shipping purposes
ShippingState	Address	Optional	The state name used for Shipping purposes
ShippingZipcode	<u>Address</u>	Optional	The zip code used for Shipping purposes
ShippingEmail	<u>Address</u>	Optional	The email address used for Shipping purposes
ShippingPhone	Address	Optional	The phone number used for Shipping purposes

² Used with Account Updater service only.

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Unique services identifier
PaymentAccountID	Response.PaymentAccount	Returned	Unique GUID that identifies the Payment Account.
PaymentAccountReferenceeNumber	Response.PaymentAccount	Returned	Payment Account Reference Number

Remarks

The required fields are dependent upon the *PaymentAccountType* selected.

If PaymentAccountType = CreditCard then exactly one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or EncryptedTrack2Data or EncryptedTrack1Data or EncryptedCardData

PaymentAccountDelete

The PaymentAccountDelete method is used to delete a Payment Account record.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
PaymentAccountID	<u>PaymentAccount</u>	Required	Unique GUID that identifies the Payment Account.

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Unique services identifier

Remarks

This method will mark a Payment Account record for deletion and will be purged from the system within twenty-four hours.

PaymentAccountUpdate

The PaymentAccountUpdate method is used to update an existing Payment Account record.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
PaymentAccountID	PaymentAccount	Required	Unique GUID that identifies the Payment Account.
PaymentAccountReferenceNumber	<u>PaymentAccount</u>	Required	Payment Account Reference Number
PaymentAccountType	PaymentAccount	Required ¹	Payment Account Type
▼ Track1Data	Card	Conditional ²	Track1 Data
▼ Track2Data	Card	Conditional ²	Track2 Data
MagneprintData	Card	Conditional ²	Magneprint Data
CardNumber	Card	Optional	Card Number
ExpirationMonth	Card	Conditional - (Required if card number is updated)	Expiration Month (MM)
ExpirationYear	Card	Conditional - (Required if card number is updated)	Expiration Year (YY)
EncryptedTrack1Data	Card	Conditional ²	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	Card	Conditional ²	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	Card	Conditional ²	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerialNumber	Card	Conditional ²	Card Data Key Serial Number from integrated encryption device
<u> EncryptedFormat</u>	Card	Optional	Encryption format from

Name	Class	Required	Description
			integrated encryption device
AccountNumber AccountNumber	DemandDepositAccount	Conditional - (Required if PaymentAccountType is a DDA account)	Account Number
RoutingNumber	<u>DemandDepositAccount</u>	Conditional - (Required if PaymentAccountType is a DDA account)	Routing Number
BillingName	<u>Address</u>	Optional	The name used for billing purposes
BillingAddress1	<u>Address</u>	Optional	The street address used for billing purposes
BillingAddress2	<u>Address</u>	Optional	The street address used for billing purposes
BillingCity	<u>Address</u>	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	<u>Address</u>	Optional	The zip code used for billing purposes
BillingEmail	Address	Optional	The email address used for billing purposes
BillingPhone	<u>Address</u>	Optional	The phone number used for billing purposes
ShippingName	<u>Address</u>	Optional	The name used for shipping purposes
ShippingAddress1	<u>Address</u>	Optional	The street address used for Shipping purposes
ShippingAddress2	Address	Optional	The street address used for Shipping purposes
ShippingCity	Address	Optional	The city name used for Shipping purposes
ShippingState	<u>Address</u>	Optional	The state name used for Shipping purposes
ShippingZipcode	<u>Address</u>	Optional	The zip code used for Shipping purposes
ShippingEmail	<u>Address</u>	Optional	The email address used for Shipping purposes
ShippingPhone	<u>Address</u>	Optional	The phone number used for Shipping purposes

¹ PaymentAccountType cannot be updated

If PaymentAccountType = CreditCard then exactly one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or EncryptedTrack2Data or EncryptedTrack1Data or EncryptedCardData

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Unique services identifier
PaymentAccountID	Response.PaymentAccount	Returned	Unique GUID that identifies the Payment Account.
PaymentAccountReferenc eNumber	Response.PaymentAccount	Returned	Payment Account Reference Number

Remarks

The required fields are dependent upon the PaymentAccountType selected.

■ PaymentAccountQuery

The *PaymentAccountQuery* method is used to query existing Payment Account record(s). The *QueryData* field will return Payment Account information in XML format, with a maximum of 1000 records returned.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
PaymentAccountID	<u>PaymentAccountParameters</u>	Optional	Unique GUID that identifies the Payment Account.
PaymentAccountReferenceNumber	<u>PaymentAccountParameters</u>	Optional	Payment Account Reference Number
PaymentBrand	<u>PaymentAccountParameters</u>	Optional	Card type query value
ExpirationMonthBegin	<u>PaymentAccountParameters</u>	Optional	Card expiration month query value
ExpirationMonthEnd	<u>PaymentAccountParameters</u>	Optional	Card expiration month query value
ExpirationYearBegin	<u>PaymentAccountParameters</u>	Optional	Card expiration year search value
ExpirationYearEnd	<u>PaymentAccountParameters</u>	Optional	Card expiration year query value
TransactionSetupID	<u>PaymentAccountParameters</u>	Optional	Transaction Setup IDs
PASSUpdaterDateTimeBegin	<u>PaymentAccountParameters</u>	Optional	Begin date/time of range formatted [yyyy-MM-dd HH:mm:ss.fff]
PASSUpdaterDateTimeEnd	<u>PaymentAccountParameters</u>	Optional	End date/time of range formatted [yyyy-MM-dd HH:mm:ss.fff]
■ PASSUpdaterBatchStatus	<u>PaymentAccountParameters</u>	Optional	Specifies whether or not the PASS record will be included in the next PASS Updater batch
PASSUpdaterStatus	<u>PaymentAccountParameters</u>	Optional	Specifies the match status of the PASS record

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
QueryData	Response	Returned	Records returned in xml format (<items><item></item> of 1000 records can be returned.</items>
ServicesID	Response	Returned	Unique services identifier
PaymentAccountID	Response.PaymentAccount	Conditional Returned if input value passed	Unique GUID that identifies the Payment Account.
PaymentAccountReferenceNumber	Response.PaymentAccount	Conditional Returned if input value passed	Payment Account Reference Number

Remarks

- This method will return the resulting query in xml format based on the input parameters passed in via the *PaymentAccountParameters* object.
- At least one *PaymentAccountParameters* value required to query records.

■ PaymentAccountAutoUpdate

The *PaymentAccountAutoUpdate* method is **only used with the TransForm Tokenization Auto Updater product.**

Input Fields

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
*AcceptorID	Credentials	Required	MerchantID
PaymentAccountID	PaymentAccount	Required	Unique GUID that identifies the Payment Account.
PASSUpdaterBatchStatus	PaymentAccount	Required	Specifies whether or not the PASS record will be included in the next PASS Updater batch
[■] PASSUpdaterOption	<u>PaymentAccount</u>	Optional	Specifies whether or not the PASS record will be updated automatically

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Unique services identifier
PaymentAccountID	Response. PaymentAccount	Returned	Unique GUID that identifies the Payment Account.

PaymentAccountCreateWithTransID

The PaymentAccountCreateWithTransID method is used to create a new Payment Account record from an existing TransactionID.

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
* AcceptorID	Credentials	Required	MerchantID
PaymentAccountType	PaymentAccount	Required	Payment Account Type
PaymentAccountReferenceNumber	<u>PaymentAccount</u>	Required	Payment Account Reference Number
TransactionID	Transaction	Required	Unique Transaction identifier
BillingName	Address	Optional	The name used for billing purposes
BillingAddress1	Address	Optional	The street address used for billing purposes
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	Address	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	Address	Optional	The zip code used for billing purposes
BillingEmail	Address	Optional	The email address used for billing purposes
BillingPhone	Address	Optional	The phone number used for billing purposes
ShippingName	Address	Optional	The name used for shipping purposes
ShippingAddress1	Address	Optional	The street address used for Shipping purposes
ShippingAddress2	<u>Address</u>	Optional	The street address used for Shipping purposes
ShippingCity	<u>Address</u>	Optional	The city name used for Shipping purposes
ShippingState	Address	Optional	The state name used for Shipping purposes

Name	Class	Required	Description
ShippingZipcode	<u>Address</u>	Optional	The zip code used for Shipping purposes
ShippingEmail	<u>Address</u>	Optional	The email address used for Shipping purposes
ShippingPhone	<u>Address</u>	Optional	The phone number used for Shipping purposes

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Unique services identifier
PaymentAccountID	Response.PaymentAcc ount	Returned	Unique GUID that identifies the Payment Account.
PaymentAccountReferenceeNumber	Response.PaymentAcc ount	Returned	Payment Account Reference Number

Remarks

• It is recommended that *TransactionID*'s **only** from the following Transaction Types be used to generate a *PaymentAccountID*:

CreditCardSale
CreditCardCredit
CreditCardAuthorization
CreditCardAVSOnly
CreditCardForce
CreditCardBalanceInquiry

DebitCardSale
DebitCardReturn
CheckSale
CheckCredit
CheckVerification

- The PaymentAccountType submitted must be correct for the associated TransactionID submitted.
- The PaymentAccountCreateWithTransID can be used to create Payment Account records from transactions as old as the Vantiv Integrated Payments Data Retention policy allows (120 days).

PaymentAccountQueryRecordCount

The PaymentAccountQueryRecordCount method is used to query the number of existing Payment Account record(s). The QueryData field will return Payment Account record count and paging information in xml format. Paging details can be used with the PaymentAccountQueryTokenReport method.

Input Fields

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Unique services identifier
QueryData	Response	Returned	Records returned in xml format (<querydata><results></results>).</querydata>
			Within <results></results> :
			RecordCount = # PASS Records
			PageCount = # of Pages

Remarks

This method will return the resulting query in xml format.

PaymentAccountQueryTokenReport

The PaymentAccountQueryTokenReport method is used to obtain details about existing Payment Account records. The QueryData field will return Payment Account record information in xml format.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
Page	<u>Paging</u>	Required	Page number of records to retrieve, e.g. 1

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
	Response	Returned	Unique services identifier
QueryData	Response	Returned	Records returned in xml format (<items><i></i></items>). Within <i></i> :
			<t></t> = PaymentAccountID
			<r></r> = PaymentAccount
			ReferenceNumber

Remarks

This method will return the resulting query in xml format.

ScheduledTaskDelete

The ScheduledTaskDelete method is used to delete an existing Scheduled Task.

™Note Accessible through the Express Services Interface

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
ScheduledTaskID	ScheduledTask	Required	Unique value that identifies the Scheduled Task.

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimeZone	Response	Returned	Express transaction UTC time zone

Remarks

This method will delete a Scheduled Task permanently.

ScheduledTaskQuery

The *ScheduledTaskQuery* method is used to query existing Scheduled Tasks. The *QueryData* field will return Scheduled Task information in xml format.

™Note Accessible through the Express Services Interface

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
Scheduled TaskID	ScheduledTaskParameters	Optional	Unique GUID that identifies the Scheduled Task.
ScheduledTaskReferenceNumber	ScheduledTaskParameters	Optional	User defined unique reference number that identifies this scheduled task.
Scheduled Task Name	ScheduledTaskParameters	Optional	User generated name of the task.
Scheduled Task Group ID	ScheduledTaskParameters	Optional	Express platform generates this.
RunStartDate	ScheduledTaskParameters	Optional	The start date. Format is YYYYMMDD.
E RunUntilCancelFlag	<u>ScheduledTaskParameters</u>	Optional	If this is set to true, the task will run every scheduled time until the task is deleted by the user.
[®] RunFrequency	ScheduledTaskParameters	2	Specifies how often an event should occur.
ScheduledTaskStatus	ScheduledTaskParameters	2	The status of the task. You can set this to Active or Disabled.
Scheduled Task Query Type	ScheduledTaskParameters	Optional	Query type used to provide additional event information (0=Default, 1=RunLog)
■ Scheduled Task Run Status	<u>ScheduledTaskParameters</u>	Optional	The run status of the task.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimeZone	Response	Returned	Express transaction UTC time zone
QueryData	Response	Returned	Records returned in xml format (<items><item></item>>) . A maximum of 1000 records can be returned.</items>
ServicesID	Response	Returned	Unique services identifier
ScheduledTaskID	Response.ScheduledTask	Returned	Unique value that identifies the Scheduled Task
Scheduled Task Reference Number	Response.ScheduledTask	Conditional Returned if input value passed	User defined unique reference number that identifies this scheduled task.

Remarks

- This method will return the resulting query in xml format based on the input parameters passed in via the *ScheduledTaskParameters* object.
- At least one ScheduleTaskParameters value required to query records

ScheduledTaskUpdate

The ScheduledTaskUpdate method is used to update an existing Scheduled Task.

☞Note Accessible through the Express Services Interface

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
* AcceptorID	Credentials	Required	MerchantID
Scheduled TaskID	ScheduledTask	Required	Unique value that identifies the Scheduled Task.
ScheduledTaskReferenceNumber	ScheduledTask	Required	
■ RunFrequency	ScheduledTask	Required	
■ RunUntilCancelFlag	ScheduledTask	Required	
RunCycles	ScheduledTask	Required only when RunUntilCancelFlag set to false	
■ ScheduledTaskStatus	ScheduledTask	Required	
ScheduledTaskName	ScheduledTask	Optional	User generated name of the task.
Scheduled Task Group ID	<u>ScheduledTask</u>	Optional	Express platform generates this.

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimeZone	<u>Response</u>	Returned	Express transaction UTC time zone

Remarks

This method will update a Scheduled Task.

ScheduledTaskRetry

The ScheduledTaskRetry method is used to rerun a single, previously non-approved Scheduled Task.

☞Note Accessible through the Express Services Interface

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	<u>Credentials</u>	Required	Secret token used for authentication
* AcceptorID	Credentials	Required	MerchantID
ScheduledTaskRunLogID	<u>ScheduledTask</u>	Optional	Run log ID of a Scheduled Task

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimeZone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Unique services identifier
TransactionID	Response. Transaction	Returned	Unique transaction identifier
ProcessorName	Response Transaction	Returned	Name of processor
TransactionStatus	Response. Transaction	Returned	Description of status/state of transaction
* TransactionStatusCode	Response Transaction	Returned	Status/State of transaction
ScheduledTaskID	Response.ScheduledTask	Returned	Unique GUID that identifies the Payment Account.

TokenCreate

The *TokenCreate* method is used to create a token for a given card.

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
CardNumber	Card	Conditional ¹	Card Number
ExpirationMonth	Card	Conditional ¹	Expiration Month (MM)
ExpirationYear	Card	Conditional ¹	Expiration Year (YY)
Track1Data	Card	Conditional ¹	Track1 Data
▼ Track2Data	Card	Conditional ¹	Track2 Data
MagneprintData	Card	Conditional ¹	Magneprint Data
EncryptedTrack1Data	<u>Card</u>	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	<u>Card</u>	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	<u>Card</u>	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerialNumber	<u>Card</u>	Conditional ¹	Card Data Key Serial Number from integrated encryption device
€ PEncryptedFormat	<u>Card</u>	Optional	Encryption format from integrated encryption device
* TokenProvider	<u>Token</u>	Required	TokenProvider enumeration values Null = 0 ExpressPASS = 1 OmniToken = 2 Paymetric = 3 TransArmor = 4

Name	Class	Required	Description
	<u>Token</u>	Required for Paymetric TokenProvider Optional for others	Merchant VaultID
* TerminalID	<u>Terminal</u>	Optional	Unique terminal identifier
■ TerminalType	Terminal	Optional	Terminal Type
TerminalCapabilityCode	Terminal	Optional	Terminal Capability Code
<u>TerminalEnvironmentCode</u>	Terminal	Optional	Terminal Environment Code
<u>■ CardPresentCode</u>	Terminal	Optional	Card Present Code
EP CVVPresenceCode	<u>Terminal</u>	Optional	Card Verification Value Presence Code
E CVVResponseType	<u>Terminal</u>	Optional	Card Verification Value Presence Code
 <u>CardInputCode</u>	Terminal	Optional	Card Input Code
CardholderPresentCode	Terminal	Optional	Cardholder Present Code
■ MotoECICode	<u>Terminal</u>	Optional	Mail Order Telephone Order Electronic Indicator Code
<u>ConsentCode</u>	Terminal	Optional	Consumer Consent Code
TerminalEncryptionFormat	<u>Terminal</u>	Optional	Not used
* TerminalSerialNumber	<u>Terminal</u>	Optional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	Terminal	Optional	Lane Number

Exactly one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or EncryptedTrack2Data or EncryptedTrack1Data or EncryptedCardData

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	<u>Response</u>	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
TruncatedCardNumber	Response.Card	Returned	Truncated card number
ExpirationMonth	Response.Card	Returned	Expiration month
ExpirationYear	Response.Card	Returned	Expiration year
© CardLogo	Response.Card	Returned	Payment brand
* TokenID	Response.Token	Returned	Token
TokenProvider	Response.Token	Returned	TokenProvider type
ServicesID	Response	Returned	Unique services identifier

TokenCreateWithTransID

The *TokenCreateWithTransID* method is used to create a token for a card from an existing *TransactionID*.

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
ApplicationVersion	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
TransactionID	Transaction	Required	Unique transaction identifier
▼ TokenProvider	<u>Token</u>	Required	TokenProvider enumeration values Null = 0 ExpressPASS = 1 OmniToken = 2 Paymetric = 3 TransArmor = 4
VaultID	<u>Token</u>	Required for Paymetric TokenProvider Optional for others	Merchant VaultID
TerminalID	<u>Terminal</u>	Optional	Unique terminal identifier
<u> TerminalType</u>	<u>Terminal</u>	Optional	Terminal Type
<u>FarminalCapabilityCode</u>	Terminal	Optional	Terminal Capability Code
■ TerminalEnvironmentCode	<u>Terminal</u>	Optional	Terminal Environment Code
[€] CardPresentCode	Terminal	Optional	Card Present Code
₽ CVVPresenceCode	<u>Terminal</u>	Optional	Card Verification Value Presence Code
<u>■</u> CVVResponseType	<u>Terminal</u>	Optional	Card Verification Value Presence Code
E CardInputCode	<u>Terminal</u>	Optional	Card Input Code
<u>CardholderPresentCode</u>	<u>Terminal</u>	Optional	Cardholder Present Code

Name	Class	Required	Description
₽ MotoECICode	<u>Terminal</u>	Optional	Mail Order Telephone Order Electronic Indicator Code
i PConsentCode	Terminal	Optional	Consumer Consent Code
TerminalEncryptionFormat	<u>Terminal</u>	Optional	Not used
* TerminalSerialNumber	<u>Terminal</u>	Optional	Terminal serial number from integrated encryption device (submit with encrypted data)
LaneNumber	Terminal	Optional	Lane Number

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
TruncatedCardNumber	Response.Card	Returned	Truncated card number
ExpirationMonth	Response.Card	Returned	Expiration month
ExpirationYear	Response.Card	Returned	Expiration year
CardLogo	Response.Card	Returned	Payment brand
▼ TokenID	Response.Token	Returned	Token
₱ TokenProvider	Response.Token	Returned	TokenProvider type
ServicesID	Response	Returned	Unique services identifier

System

HealthCheck

The *HealthCheck* method is the simplest method and is used to both validate communication integrity and determine the availability of the host processing network. A successful response from a *HealthCheck* request does not imply or ensure the success of any subsequent transactions.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID AccountID	Credentials	Required	Unique account identifier
AccountToken	<u>Credentials</u>	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Returned	Host Response Code
HostResponseMessage	Response	Returned	Host Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	<u>Response</u>	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone

AccountTokenCreate

The AccountTokenCreate method is used to generate a new AccountToken that will replace an existing active AccountToken. In order to activate and use this new AccountToken for authentication, AccountTokenActivate must be called.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
NewAccountToken	Credentials	Returned	New secret token used for authentication
ProcessorName	Transaction	Returned	Name of processor

AccountTokenActivate

The *AccountTokenActivate* method is used to disable the previous AccountToken and activate the new one that was generated when calling *AccountTokenCreate*.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	MerchantID
NewAccountToken	Credentials	Required	New secret token used for authentication

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ProcessorName	Transaction	Returned	Name of processor

Hosted Payments

TransactionSetup

The *TransactionSetup* method is used in conjuction with *Express* Hosted Payments transaction processing.

Input Fields

Name	Class	Required	Description
ApplicationID	<u>Application</u>	Required	Unique application identifier
ApplicationName	<u>Application</u>	Required	Name of application
Application Version	Application	Required	Version of application
* AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
AcceptorID	Credentials	Required	MerchantID
▼ <u>TransactionAmount</u>	<u>Transaction</u>	Conditional	Amount of transaction (for AVS Only method, use 0.00)
ClerkNumber	Transaction	Optional	Clerk number
ShiftID	Transaction	Optional	Shift ID
SalesTaxAmount	Transaction	Optional	Sales tax amount
[®] TipAmount	Transaction	Optional	Tip amount
ConvenienceFeeAmount	Transaction	Optional	Convenience fee amount (Vantiv only)
ReferenceNumber	Transaction	Optional	Unique value assigned by the transaction sender
<u>▼ TicketNumber</u>	Transaction	Optional ²	Unique numeric value assigned by the transaction sender
■ MarketCode	<u>Transaction</u>	Conditional ⁵	Market Code
ApprovalNumber ApprovalNumber	<u>Transaction</u>	Conditional Needed only for Force transaction	Original approval number. Provided during voice authorization. Capital letters and numbers only.
Commercial Card Customer Code	Transaction	Optional	Commercial card customer code
▼ TerminalID	Terminal	Required	Unique terminal identifier
■ PartialApprovedFlag	Transaction	Conditional ⁷	Indicates whether software supports partial approvals
MerchantVerificationValue	Transaction	Conditional ⁷	Healthcare Merchant Verification Value
<u>■ CardPresentCode</u>	Terminal	Conditional ⁵	Card Present Code
<u>CardholderPresentCode</u>	<u>Terminal</u>	Conditional ⁵	Cardholder Present Code
[₽] CardInputCode	<u>Terminal</u>	Conditional ⁵	Card Input Code

Name	Class	Required	Description
<u>CVVPresenceCode</u>	<u>Terminal</u>	Conditional ⁵	Card Verification Value Presence Code
■ TerminalCapabilityCode	Terminal	Conditional ⁵	Terminal Capability Code
TerminalEnvironmentCode	<u>Terminal</u>	Conditional ⁵	Terminal Environment Code
[₽] MotoECICode	<u>Terminal</u>	Conditional ⁵	Mail Order Telephone Order Electronic Indicator Code
TerminalType	Terminal	Optional	Terminal Type
TerminalEncryptionFormat	Terminal	Optional	Not used
* TerminalSerialNumber	<u>Terminal</u>	Conditional	Terminal serial number from integrated encryption device (submit with encrypted data)
<u>LaneNumber</u>	Terminal	Conditional ⁹	Lane number
PaymentAccountID	PaymentAccount	Conditional Needed only for Payment Account Update transaction	Payment Account ID
PaymentAccountType	<u>PaymentAccount</u>	Conditional Needed only for Payment Account Create and Update transaction	Payment Account Type
PaymentAccountReferenceNumber	<u>PaymentAccount</u>	Conditional Needed only for Payment Account Create and Update transaction	Payment Account Reference Number
BillingName	Address	Optional	The name used for billing purposes
BillingAddress1	Address	Conditional Needed for AVS	The street address used for billing purposes
BillingAddress2	Address	Optional	The street address used for billing purposes
BillingCity	Address	Optional	The city name used for billing purposes
BillingState	Address	Optional	The state name used for billing purposes
BillingZipcode	Address	Conditional Needed for AVS	The zip code used for billing purposes
BillingEmail	<u>Address</u>	Optional	The email address used for billing purposes

Name	Class	Required	Description
BillingPhone	Address	Optional	The phone number used for billing purposes
ShippingName	Address	Optional	The name used for billing purposes
ShippingAddress1	<u>Address</u>	Optional	The street address used for Shipping purposes
ShippingAddress2	Address	Optional	The street address used for Shipping purposes
ShippingCity	Address	Optional	The city name used for Shipping purposes
ShippingState	Address	Optional	The state name used for Shipping purposes
ShippingZipcode	Address	Optional	The zip code used for Shipping purposes. Recommend submitting for Level II processing.
ShippingEmail	Address	Optional	The email address used for Shipping purposes
ShippingPhone	Address	Optional	The phone number used for Shipping purposes
TransactionSetupMethod	TransactionSetup	Required	Transaction method used
<u> Device</u>	<u>TransactionSetup</u>	Optional	Entry device used
<u>Embedded</u>	TransactionSetup	Required	Specifies if card entry page will be embedded in an application
[©] CVVRequired	TransactionSetup	Optional	Requires user to enter CVV2, CVC2, or CID values
₽ AutoReturn	TransactionSetup	Optional ⁴	Specifies if approval page will be automatically directed to ReturnURL
<u>PDeviceInputCode</u>	TransactionSetup	Optional	Specifies the entry device input type
CompanyName	TransactionSetup	Optional ⁴	Name of merchant
LogoURL	TransactionSetup	Optional ⁴	HTTPS URL of merchant logo to be displayed
* Tagline	TransactionSetup	Optional ⁴	Text displayed next to logo
WelcomeMessage	<u>TransactionSetup</u>	Optional ⁴	Text to display to customer (if not provided, generic message will be displayed)
ReturnURL	TransactionSetup	Optional ⁴	Return URL link displayed after transaction completion or cancelled
ReturnURLTitle	<u>TransactionSetup</u>	Optional ⁴	Text to display for ReturnURL (if not provided generic "return to merchant" message will be

Name	Class	Required	Description
			displayed
OrderDetails	TransactionSetup	Optional ⁴	Text to describe purchase or detail information
ProcessTransactionTitle	TransactionSetup	Optional	Text to describe submit button
LodgingAgreementNumber	ExtParm.Lodging	Conditional ⁶	Agreement (Folio) number
LodgingCheckInDate	ExtParm.Lodging	Conditional ⁶	Check-in date
LodgingCheckOutDate	ExtParm.Lodging	Conditional ⁶	Check-out date
LodgingRoomAmount	ExtParm.Lodging	Conditional ⁶	Nightly rate for one room
LodgingRoomTax	ExtParm.Lodging	Conditional ⁶	Reserved for future use
LodgingNoShowIndicator	ExtParm.Lodging	Conditional ⁶	No-show indicator
LodgingDuration	ExtParm.Lodging	Conditional ⁶	Number of days
LodgingCustomerName	ExtParm.Lodging	Conditional ⁶	Customer name
LodgingClientCode	ExtParm.Lodging	Conditional ⁶	Reserved for future use
LodgingExtraChargesDetail	ExtParm.Lodging	Conditional ⁶	Extra charges detail
LodgingExtraChargesAmounts	ExtParm.Lodging	Conditional ⁶	Reserved for future use
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Conditional ⁶	Prestigious property code
LodgingSpecialProgramCode	ExtParm.Lodging	Conditional ⁶	Special program code
LodgingChargeType	ExtParm.Lodging	Conditional ⁶	Charge type
HealthcareFlag	ExtParm.Healthcare	Conditional ⁷	If this is set to True, the Healthcare parameters should be submitted.
#HealthcareFirstAccountType	ExtParm.Healthcare	Conditional ⁷	Healthcare account type
HealthcareFirstAmountType	ExtParm.Healthcare	Conditional ⁷	Healthcare amount type
HealthcareFirstCurrencyCode	ExtParm.Healthcare	Conditional ⁷	Currency Code. Use 840 for U.S.
₱ HealthcareFirstAmountSign	ExtParm.Healthcare	Conditional ⁷	Positive or negative amount
HealthcareFirstAmount	ExtParm.Healthcare	Conditional ⁷	Healthcare amount
HealthcareSecondAccountType	ExtParm.Healthcare	Conditional ⁷	Healthcare account type
HealthcareSecondAmountType	ExtParm.Healthcare	Conditional ⁷	Healthcare amount type
HealthcareSecondCurrencyCode	ExtParm.Healthcare	Conditional ⁷	Currency Code. Use 840 for U.S.
HealthcareSecondAmountSign	ExtParm.Healthcare	Conditional ⁷	Positive or negative amount
HealthcareSecondAmount	ExtParm.Healthcare	Conditional ⁷	Healthcare amount
HealthcareThirdAccountType	ExtParm.Healthcare	Conditional ⁷	Healthcare account type
HealthcareThirdAmountType	ExtParm.Healthcare	Conditional ⁷	Healthcare amount type
HealthcareThirdCurrencyCode	ExtParm.Healthcare	Conditional ⁷	Currency Code. Use 840 for U.S.
HealthcareThirdAmountSign	ExtParm.Healthcare	Conditional ⁷	Positive or negative amount
HealthcareThirdAmount	ExtParm.Healthcare	Conditional ⁷	Healthcare amount
HealthcareFourthAccountType	ExtParm.Healthcare	Conditional ⁷	Healthcare account type

Name	Class	Required	Description
HealthcareFourthAmountType	ExtParm.Healthcare	Conditional ⁷	Healthcare amount type
HealthcareFourthCurrencyCode	ExtParm.Healthcare	Conditional ⁷	Currency Code. Use 840 for U.S.
HealthcareFourthAmountSign	ExtParm.Healthcare	Conditional ⁷	Positive or negative amount
HealthcareFourthAmount	ExtParm.Healthcare	Conditional ⁷	Healthcare amount
AutoRentalAgreementNumber	ExtParm.AutoRental	Conditional ⁸	Rental agreement number
AutoRentalNoShowIndicator	ExtParm.AutoRental	Conditional ⁸	No show indicator
AutoRentalExtraChargesDetail	ExtParm.AutoRental	Conditional ⁸	Extra charges detail
AutoRentalPickupDate	ExtParm.AutoRental	Conditional ⁸	Rental pickup date. Format is YYYYMMDD.
AutoRentalCustomerName	ExtParm.AutoRental	Conditional ⁸	Name of the person or business with the reservation
AutoRentalReturnCity	ExtParm.AutoRental	Conditional ⁸	City where vehicle was returned
AutoRentalReturnState	ExtParm.AutoRental	Conditional ⁸	State where vehicle was returned
AutoRentalReturnLocationID	ExtParm.AutoRental	Conditional ⁸	Rental location ID
AutoRentalDuration	ExtParm.AutoRental	Conditional ⁸	Number of rental days (e.g. 03). Pass 01 for no-show rentals.
AutoRentalPickupLocation	ExtParm.AutoRental	Conditional ⁸	Name of business where vehicle was picked up
AutoRentalPickupCity	ExtParm.AutoRental	Conditional ⁸	Name of city, town, or village where vehicle was picked up
AutoRentalPickupState	ExtParm.AutoRental	Conditional ⁸	Abbreviation of state, province, or other country subdivision where vehicle was picked up
<u>AutoRentalPickupCountryCode</u>	ExtParm.AutoRental	Conditional ⁸	Country code where vehicle was picked up. For US, use 840.
AutoRentalPickupTime	ExtParm.AutoRental	Conditional ⁸	Local time when vehicle was picked up.
AutoRentalReturnCountryCode	ExtParm.AutoRental	Conditional ⁸	Country code where vehicle was returned. For US, use 840.
AutoRentalReturnDate	ExtParm.AutoRental	Conditional ⁸	Date when vehicle was returned. Format YYYYMMDD.
<u>AutoRentalReturnTime</u>	ExtParm.AutoRental	Conditional ⁸	Local time when vehicle was returned
<u>AutoRentalVehicleClassCode</u>	ExtParm.AutoRental	Conditional ⁸	Classification of rental vehicle
<u>AutoRentalDistance</u>	ExtParm.AutoRental	Conditional ⁸	Distance traveled during rental period
² AutoRentalDistanceUnit	ExtParm.AutoRental	Conditional ⁸	Unit of measurement applicable to distance traveled
AutoRentalAuditAdjustmentCode	ExtParm.AutoRental	Conditional ⁸	Indicates if adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement
AutoRentalAuditAdjustmentAmount	ExtParm.AutoRental	Conditional ⁸	Adjustment amount for mileage, fuel, vehicle damage, etc. after a

Name	Class	Required	Description
			vehicle was returned

- ¹ Required when Scheduled Task object is being included.
- ² Used for Direct Marketing, MOTO, and E-Commerce transactions. Required when card number is manually keyed.
- ³ Used when desiring to associate a task to an already-existing *ScheduledTaskGroupID*.
- Used when Embedded flag is set to False.
- Required on all requests except when creating or updating payment account records.
- ⁶ Used when submitting a Lodging transaction (TSYS and FDC platform only). If TransactionType is *CreditCardSale* or *CreditCardForce*, all Lodging fields are required. If TransactionType is *CreditCardAuthorization*, only *LodgingDuration* is required.
- ⁷ Submitted if partial approvals are supported by integrated application.
- Used when submitting an Auto Rental transaction (TSYS and FDC platform only). If TransactionType is CreditCardSale or CreditCardForce, all Auto Rental fields are required. If TransactionType is CreditCardAuthorization, only AutoRentalDuration is required.
- ⁹ Lane number is required for some processors; including the lane number is never an error.

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code.
ExpressResponseMessage	Response	Returned	Express Response Message.
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
* TransactionSetupID	Response Transaction	Returned	Transaction Setup ID
TransactionSetupID	Response. Payment Account	Returned	Transaction Setup ID
* TransactionSetupID	Response TransactionSetup	Returned	Transaction Setup ID
	Response TransactionSetup	Returned	Validation Code

TransactionSetupExpire

The *TransactionSetupExpire* method is used to force a *TransactionSetupID* to expire. Once expired, it can no longer be used.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	<u>Application</u>	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
* AcceptorID	Credentials	Required	MerchantID
TransactionSetupID	TransactionSetup	Required	Transaction Setup ID

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code.
ExpressResponseMessage	Response	Returned	Express Response Message.
ExpressTransactionDate	Response	Returned	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime	Response	Returned	Express transaction time formatted [HHMMSS]
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone

BIN Management

Express provides two methods to determine if a card falls into a specific BIN range: BINQuery and EnhancedBINQuery. The EnhancedBINQuery method provides the most detailed BIN information and is recommended for all new development. The BINQuery method is maintained for existing applications. The input fields are the same for both methods; the output fields are different.

BINQuery

The *BINQuery* method is used to determine if a card falls into a specific BIN range, such as Debit or Healthcare.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
Application Version	Application	Required	Version of application
* AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Token used for authentication
* AcceptorID	Credentials	Required	MerchantID
CardNumber	Card	Conditional ¹	Account number on card
ExpirationMonth	Card	Conditional ¹	Expiration month
ExpirationYear	Card	Conditional ¹	Expiration year
Track1Data	<u>Card</u>	Conditional ¹	Track 1 data
Track2Data	Card	Conditional ¹	Track 2 data
MagneprintData	<u>Card</u>	Conditional ¹	Magneprint data
EncryptedTrack1Data	<u>Card</u>	Conditional ¹	Encrypted Track 1 Data from integrated encryption device
EncryptedTrack2Data	<u>Card</u>	Conditional ¹	Encrypted Track 2 Data from integrated encryption device
EncryptedCardData	<u>Card</u>	Conditional ¹	Encrypted Card Data (keyed) from integrated encryption device
CardDataKeySerialNumber	<u>Card</u>	Conditional ¹	Card Data Key Serial Number from integrated encryption device
■ EncryptedFormat	<u>Card</u>	Optional	Encryption format from integrated encryption device
PaymentAccountID	ExtParm.PaymentAccount	Conditional ¹	A unique system generated identifier.
TerminalID	Terminal	Conditional ²	Unique terminal identifier

Only one of the following field groups needs to be included: CardNumber / ExpirationMonth / ExpirationYear or Track1Data or Track2Data or MagneprintData or PaymentAccountID or EncryptedTrack1Data or EncryptedTrack2Data or EncryptedCardData.

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code.
ExpressResponseMessage	Response	Returned	Text response message
ExpressTransactionDate	Response	Returned	Express platform transaction date
ExpressTransactionTime	Response	Returned	Express platform transaction
ExpressTransactionTimezone	Response	Returned	Express platform transaction time zone
BINTypeCode	Response.BIN	Returned	Numeric BIN type code
BINTypeValue	Response.BIN	Returned	BIN type value
BINDecorator	Response.BIN	Returned if BIN is found	BIN decorator

EnhancedBINQuery

The EnhancedBINQuery method provides detailed BIN information and returns a variable number of fields, each representing a single BIN attribute, such as PrepaidCard, DebitCard, and DurbinBINRegulation.

Input Fields

The *EnahancedBINQuery* method uses the same input fields as the *BINQuery* method, described in section 0.

Output Fields

The EnhancedBINQuery method returns only those BIN indicators which apply to the card queried.

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code.
ExpressResponseMessage	Response	Returned	Text response message
ExpressTransactionDate	Response	Returned	Express platform transaction date
ExpressTransactionTime	Response	Returned	Express platform transaction
ExpressTransactionTimezone	Response	Returned	Express platform transaction time zone
Status	Response.EnhancedBIN	Returned ¹	Query status
	Response.EnhancedBIN	Returned ²	CheckCard indicator
CommercialCard	Response.EnhancedBIN	Returned ²	CommercialCard indicator
	Response.EnhancedBIN	Returned ²	CreditCard indicator
DebitCard	Response.EnhancedBIN	Returned ²	DebitCard indicator
DurbinBINRegulation	Response.EnhancedBIN	Returned ³	DurbinBINRegulation
₽ EBT	Response.EnhancedBIN	Returned ²	EBT indicator

² Used only when submitting request on TSYS and FDC platform.

Name	Class	Returned	Description
	Response.EnhancedBIN	Returned ²	FleetCard indicator
GiftCard	Response.EnhancedBIN	Returned ²	GiftCard indicator
HSAFSACard	Response.EnhancedBIN	Returned ²	HSAFSACard indicator
InternationalBIN	Response.EnhancedBIN	Returned ²	InternationalBIN indicator
PinLessBillPay	Response.EnhancedBIN	Returned ²	PinLessBillPay indicator
PrepaidCard	Response.EnhancedBIN	Returned ²	PrepaidCard indicator
	Response.EnhancedBIN	Returned ²	WIC indicator

returned as "Found" or "NotFound" according to whether BIN information is included in the response returned with value "Y" if applicable to queried card, otherwise not returned returned with integer indicating the type of Durbin regulation for queried card

Classes

This seection describes the List of Classes used with Express

Address Class

Name	Value	Max Length	Description
BillingName	Any string	100	The name used for billing purposes
BillingEmail	Any valid email address someone@somewhere.com	80	The email address used for billing purposes
BillingPhone	Recommended format (123)123-1234	20	The phone number used for billing purposes
BillingAddress1	Any string	50	The street address used for billing purposes
BillingAddress2	Any string	50	The street address used for billing purposes
BillingCity	Any string	40	The name of the city used for billing purposes
BillingState	Any two digit state code or full state name	30	The name of the state used for billing purposes
BillingZipcode	Any string	20	The zip code used for billing purposes
ShippingName	Any string	100	The name used for shipping purposes
ShippingEmail	Any valid email address someone@somewhere.com	80	The email address used for shipping purposes
ShippingPhone	Recommended format (123)123-1234	20	The phone number used for shipping purposes
ShippingAddress1	Any string	50	The street address used for shipping purposes
ShippingAddress2	Any string	50	The street address used for shipping purposes
ShippingCity	Any string	40	The name of the city used for shipping purposes
ShippingState	Any two digit state code or full state name	30	The name of the state used for shipping purposes
ShippingZipcode	Any string	20	The zip code used for shipping purposes

Application Class

The Application Class is required for every transaction and is used to uniquely identify the sending application. During certification a unique application ID will be assigned for the purpose of tracking all certified transaction types. Only an application's certified transactions types will be allowed in the production environment.

Field Names

Name	Value	Max Length	Description
ApplicationID	Valid application ID issued by Element	40	Unique application identifier
ApplicationName	Any string	50	Name of application
ApplicationVersion	Required format #.#.# example 1.2.1	50	Version of application

AutoRental Class

The *AutoRental* Class is required for Auto Rental industry transactions and is used to identify auto rental transaction details. This method is used on the TSYS or FDC platform only.

Name	Value	Max Length	Description
AutoRentalAgreementNumber	Auto Rental agreement number	14	Auto Rental agreement number
AutoRentalNoShowIndicator	No-show indicator	1	Indicates whether rental vehicle was reserved, but not actually rented. 0 = Not Applicable 1 = No Show
AutoRentalExtraChargesDetail	Auto Rental extra charges	6	Indicates any additional ancillary charges. Field can contain multiple indicators. 0 = No Extra Charge 1 = Gasoline 2 = Extra Mileage 3 = Late Return 4 = One Way Service Fee 5 = Parking or Moving Violation
AutoRental Pickup Date	Rental pickup date	8	Auto Rental pickup date. Format is YYYYMMDD.
AutoRentalDropoffDate	Reserved for future use	8	Reserved for future use.
AutoRentalCustomerName	Customer name	20	Auto Rental customer name
AutoRentalReturnCity	City	18	Auto Rental return city
AutoRentalReturnState	State	3	Auto Rental return state

Name	Value	Max Length	Description
AutoRentalReturnLocationID	Location ID	10	Auto Rental location ID where vehicle was returned
AutoRentalDuration	Number of rental days (e.g. 03). Pass 01 for no-show rentals.	2	Number of rental days (e.g. 03). Pass 01 for no-show rentals.
AutoRentalPickupLocation	Business location	38	Name of business where vehicle was picked up
AutoRentalPickupCity	City	18	Name of city, town, or village where vehicle was picked up
AutoRentalPickupState	State	3	Abbreviation of state, province, or other country subdivision where vehicle was picked up
AutoRentalPickupCountryCode	Country	3	Country code where vehicle was picked up. For US, use 840.
AutoRentalPickupTime	Pickup time	6	Local time when vehicle was picked up. Format HHMMSS.
AutoRentalReturnCountryCode	Country	3	Country code where vehicle was returned. For US, use 840.
AutoRentalReturnDate	Return date	8	Date when vehicle was returned. Format YYYYMMDD.
AutoRentalReturnTime	Return time	6	Local time when vehicle was returned. Format HHMMSS.
■ AutoRentalVehicleClassCode	Vehicle class ID	2	Classification of rental vehicle
AutoRentalDistance	Distance traveled	5	Numeric distance traveled during rental period
AutoRentalDistanceUnit	Unit of measurement	2	Unit of measurement applicable to distance traveled
AutoRentalAuditAdjustmentCode	Adjustment code	2	Indicates if adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement
AutoRentalAuditAdjustmentAmount	Adjustment amount	12	Adjustment amount for mileage, fuel, vehicle damage, etc. after a vehicle was returned

* Batch Class

Name	Value	Max Length	Description
HostBatchID	Any valid host batch number	25	Unique host batch identifier
MostItemID	Any valid host batch item number	25	Unique host batch item identifier
■ BatchCloseType	Any valid batch close type	2	Type of batch close
HostBatchItems		N/A	
■ BatchQueryType	Any valid batch query type	2	Batch query type
HostBatchCount	Any valid host batch count	5	Batch count
HostBatchAmount	Any valid host batch amount	12	Batch amount
HostCreditSaleCount	Any valid host credit sale count	5	Total numeric count of credit sales
HostCreditSaleAmount	Any valid host credit sale amount	12	Total dollar amount of credit sales with no decimal place: example \$21.52 would be 2152
HostCreditReturnCount	Any valid host credit return count	5	Total numeric count of credit returns
HostCreditReturnAmount	Any valid host credit return amount	12	Total dollar amount of credit returns with no decimal place: example \$21.52 would be 2152
Host Debit Sale Count	Any valid host debit sale count	5	Total numeric count of debit sales
Host Debit Sale Amount	Any valid host debit sale amount	12	Total dollar amount of debit sales with no decimal place: example \$21.52 would be 2152
HostDebitReturnCount	Any valid host debit return count	5	Total numeric count of debit returns
HostDebitReturnAmount	Any valid host debit return amount	12	Total dollar amount of credit returns with no decimal place: example \$21.52 would be 2152
BatchGroupingCode	Reserved for future use	2	See enum description
BatchIndexCode	Any valid batch index code	2	See enum description
HostReversalQueueID		25	Reversal queue ID

SIN Class

The BIN class is used when performing BINQuery requests.

Field Names

Name	Value	Max Length	Description
₱ BINTypeCode	BIN type code	4	Numeric BIN type code
71.			0 = NotFound
			1 = Debit
			2 = Healthcare
			3 = Commercial
			4 = Other
			5 = PrepaidDebit
BINTypeValue	BIN type value	75	Text associated with the numeric
, ,,,, ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,			BIN type code
BINDecorator	BIN decorator	75	Unique value defined by merchant
			or vendor and maps to BIN type
			value

Card Class

The Card Class is used to pass cardholder information.

Name	Value	Max Length	Description
CardNumber	Any valid credit card number	30	Numeric number displayed on front of card
ExpirationMonth	Two digit month MM	2	Two digit expiration month displayed on front of card e.g. 02
ExpirationYear	Two digit year YY	2	Two digit expiration month displayed on front of card e.g. 08
CardholderName	CardHolder name	80	The name of the Card holder as printed on the front of the card.
Track1Data	Any valid track 1 data string	76	The Track1 data embedded in the magnetic strip on the back of the card
▼ Track2Data	Any valid track 2 data string	37	The Track2 data embedded in the magnetic strip on the back of the card
Track3Data	Any valid track 3 data string	Unknown	Reserved for future use.
	Any valid Magneprint data string	700	The encrypted track data sent from a USB Magtek Magneprint swipe reader with encryption. Do not parse, as encrypted format may vary.
PINBlock	Any valid pin block	80	The encrypted pin block information sent from a pin

Name	Value	Max Length	Description
			encryption device
AVSResponseCode	Any valid AVS response code	2	AVS response from the Authorization request.
	Any valid CVV	4	The Card Verification Value found on the card: Visa CVV2, MasterCard CVC2, Discover CID
	Any valid CVV response code	2	CVV response from the Authorization request
	Any valid CAVV	60	The credit Cardholder Authentication Verification Value
	Any valid CAVV response code	10	CAVV response from the Authorization request
₽ XID	Any valid XID	60	Verified by Visa value
KeySerialNumber	Any valid debit Key Serial Number	20	Debit Key Serial Number
CardLogo	Payment brand	20	Payment brand. Possible values include: "Visa", "Mastercard", "Discover", "Amex", "Diners Club", "JCB", "Carte Blanche", "Other"
FincryptedTrack1Data	Any valid encrypted Track 1 data string	300	Encrypted Track 1 Data
EncryptedTrack2Data	Any valid encrypted Track 2 data string	200	Encrypted Track 2 Data
FncryptedCardData	Any valid encrypted card data string	200	Encrypted Card Data (keyed)
CardDataKeySerialNumber	Any valid card data key serial number	26	Used when any Encrypted Track or Encrypted Card fields are used
<u>EncryptedFormat</u>	Any valid EncryptedFormat enumeration	2	Encrypted Format
▼ TruncatedCardNumber ■ Truncat	Truncated card number	30	Truncated card number

Remarks

- Only CardNumber, Track1Data, Track2Data, MagneprintData, EncryptedTrack1Data, EncryptedTrack2Data, EncryptedCardData, or PaymentAccountID are required for any given transaction.
- If more than one field is populated they will be given the following order of precedence: MagneprintData, EncryptedTrack2Data, EncryptedTrack1Data, EncryptedCardData, Track2Data, Track1Data, PaymentAccountID, CardNumber.
- To avoid unintended results only populate one field per transaction.

Credentials Class

The *Credentials Class* is required for every transaction and is used to authenticate the sender. *AccountID* and *AccountToken* are assigned by Element.

Field Names

Name	Value	Max Length	Description
AccountID	Valid account ID issued by Element	10	Unique account identifier
AccountToken	Valid account token issued by Element	140	Secret token used for authentication
AcceptorID	Valid merchant ID or account number issued by host or financial institution	50	Unique merchant account identifier, most likely a Merchant ID
NewAccountToken	New valid account token issued by Element	140	New secret token used for authentication

* DemandDepositAccount Class

Field Names

Name		Value	Max Length	Description
₽ _{DD/}	AAccountType	Type of Account	2	Type of DDA Account. Checking, Savings, etc.
Acc	ountNumber	Financial Institution Account Number	20	The Account Number
₽ Rou	tingNumber	Financial Institution Routing Number	9	The Routing Number of the financial institution
Che	ckNumber	Check Number	8	Check Number
₽ Che	ckType	Type of Check	2	Type of check. Business / Personal, etc.
•	TruncatedAccount Number	Masked Account Number	20	Masked Return Value: used for display purposes.
•	TruncatedRouting Number	Masked Routing Number	9	Masked Return Value: used for display purposes.

Remarks

This class is used for passing in *DemandDepositAccount* information (i.e. Checking Account information.)

* ExtendedParameters Class

Field Names

Name	Value	Max Length	Description
Key			Name of Key
🕏 Value			Value of Key

Remarks

When not used, pass in null value for Web Service interface.

* Healthcare Class

The Healthcare Class should be submitted for Visa and MasterCard Healthcare/FSA cards only.

Name	Value	Max Length	Description
₽ HealthcareFlag	Boolean Type	1	If this is set to True, the Healthcare parameters should be submitted.
HealthcareFirstAccountType		2	
HealthcareFirstAmountType		2	
HealthcareFirstCurrencyCode		3	Currency Code. Use 840 for U.S.
HealthcareFirstAmountSign		2	0 = Positive amount1 = Negative amount
HealthcareFirstAmount		12	Healthcare amount
HealthcareSecondAccountType		2	
HealthcareSecondAmountType		2	
HealthcareSecondCurrencyCode		3	Currency Code. Use 840 for U.S.
HealthcareSecondAmountSign		2	0 = Positive amount
			1 = Negative amount
HealthcareSecondAmount		12	Healthcare amount
<u>HealthcareThirdAccountType</u>		2	
HealthcareThirdAmountType		2	
HealthcareThirdCurrencyCode		3	Currency Code. Use 840 for U.S.
HealthcareThirdAmountSign		2	0 = Positive amount 1 = Negative amount
HealthcareThirdAmount		12	Healthcare amount
HealthcareFourthAccountType		2	
HealthcareFourthAmountType		2	

Name	Value	Max Length	Description
HealthcareFourthCurrencyCode		3	Currency Code. Use 840 for U.S.
■ HealthcareFourthAmountSign		2	0 = Positive amount
			1 = Negative amount
HealthcareFourthAmount		12	Healthcare amount

☆ Identification Class

Field Names

Name	Value	Max Length	Description
BirthDate	Date of Birth	8	Format: YYYYMMDD
Drivers License Number	Drivers License Number	40	The Drivers License Number
DriversLicenseState	Drivers License State	2	2 Digit state indicator of where the license was issued.
▼ TaxIDNumber	SSN or Tax ID	9	No dashes
HealthcareFirstAccountType			Healthcare account type
HealthcareFirstAmountType			Healthcare amount type
HealthcareSecondAccountType			Healthcare account type
HealthcareSecondAmountType			Healthcare amount type
HealthcareThirdAccountType			Healthcare account type
HealthcareThirdAmountType			Healthcare amount type
HealthcareFourthAccountType			Healthcare account type
•			Healthcare amount type
HealthcareFourthAmountType			

Remarks

Be sure to format the <code>BirthDate</code> correctly.

♦ Lodging Class

The *Lodging* Class is required for **Hotel/Lodging industry** transactions and is used to identify lodging transaction details. **This method is used on the TSYS or FDC platform only**.

Name	Value	Max Length	Description
LodgingAgreementNumber	Hotel folio number	25	Hotel Folio number
LodgingCheckInDate	Check-in date	8	Check-in date. Format is YYYYMMDD. Advance check-in = date of reservation No-show = anticipated arrival date
LodgingCheckOutDate	Check-out date	8	Check-out date. Format is YYYYMMDD.
LodgingRoomAmount	Nightly rate for one room	12	Nightly rate for one room. Must always be greater than 0.00 for room reservation.
LodgingRoomTax	Room tax	12	Reserved for future use
LodgingNoShowIndicator	No-show indicator	1	Indicates a room was reserved, but not actually rented. 0 = Not Applicable 1 = No Show
LodgingDuration	Number of hotel stay days (e.g. 03).	2	Number of hotel stay days (e.g. 03). Pass 01 for no-show. From 01 to 99.
LodgingCustomerName	Customer name	26	Lodging customer name
LodgingClientCode	Client code	25	Reserved for future use
VodgingExtraChargesDetail Control Co	Extra charges detail	6	Indicates any additional ancillary charges. Field can contain multiple indicators. 0 = Not Used 1 = Reserved 2 = Restaurant 3 = Gift Shop 4 = Mini-Bar 5 = Telephone 6 = Other 7 = Laundry
•	Extra charges	250	Reserved for future use
LodgingExtraChargesAmounts	amount	2	Drastigious property sada
LodgingPrestigiousPropertyCode	Prestigious property code (Visa)	2	Prestigious property code
■ LodgingSpecialProgramCode	Special program code (American Express)	2	Special program code
<u>■ LodgingChargeType</u>	Lodging Charge Type	2	Lodging charge type

Paging Class

The Paging Class is optional with the PaymentAccountQueryTokenReport method.

Field Names

Name	Value	Max Length	Description
Page	Numeric	9	PASS Token Report page number.
· ·			e.g. 1, 2, 3,

Parameters Class

Name	Value	Max Length	Description
		30	Begin date/time of transaction range formatted [yyyy-MM-dd HH:mm:ss.fff]
TransactionDateTimeEnd		30	End date/time of transaction range formatted [yyyy-MM-dd HH:mm:ss.fff]
TransactionID		10	Unique transaction identifier
▼ TerminalID		40	Unique terminal identifier
ApplicationID		40	Unique application identifier
ApprovalNumber		30	Issuer assigned approval number
ApprovedAmount		10	Approved transaction amount
ExpressTransactionDate		30	Express transaction date formatted [YYYYMMDD]
ExpressTransactionTime		30	Express transaction time formatted [HHMMSS]
HostBatchID		25	Unique host batch identifier
HostItemID		25	Unique host batch item identifier
HostReversalQueueID		25	Reversal queue identifier
Original Authorized Amount		10	Original dollar amount authorized
ReferenceNumber		50	User defined reference number
ShiftID		10	Shift identifier
SourceTransactionID		10	Unique transaction identifier from a previous transaction
TerminalType		2	Type of terminal
* TrackingID		60	Internal transaction tracking identifier
TransactionAmount		12	Dollar amount of transaction
TransactionSetupID	GUID	50	Unique GUID that identifies the Transaction Setup ID. The Express

Name	Value	Max Length	Description
			Platform generates this.
TransactionStatus		60	Status of transaction
TransactionType		60	Type of transaction
		60	Verify by Visa value
ReverseOrder		2	Flag to query records in descending order

Remarks

Used for reporting on transactions.

* PaymentAccount Class

Field Names

Name	Value	Max Length	Description
PaymentAccountID		50	Unique GUID that identifies the Payment Account. The <i>Express</i> Platform generates this.
PaymentAccountType		2	The type of Payment Account.
PaymentAccountReference eNumber		50	User generated Reference Number used to identify a Payment Account.
*TransactionSetupID	GUID	50	Unique GUID that identifies the Transaction Setup ID. The <i>Express</i> Platform generates this.
[■]PASSUpdaterBatchStatus		2	Specifies whether or not the PASS record will be included in the next PASS Updater batch
■ PASSUpdaterOption		2	Specifies the match status of the PASS record

Remarks

Used in transactions utilizing our TransForm Tokenization service.

** PaymentAccountParameters Class

Field Names

Name	Value	Max Length	Description
PaymentAccountID		50	Unique GUID that identifies the Payment Account. The <i>Express</i> Platform generates this.
■ PaymentAccountType		2	The type of Payment Account.
PaymentAccountReference eNumber		50	User generated Reference Number used to identify a Payment Account.
PaymentBrand		50	Card type quey value. Must match exactly, and can be: Visa, MasterCard, Discover, Amex, or Diners Club.
ExpirationMonthBegin	Two-digit month	2	Card expiration month query value, e.g. 02
ExpirationMonthEnd	Two-digit month	2	Card expiration month query value, e.g. 02
ExpirationYearBegin	Two-digit year	2	Card expiration year query value, e.g. 09
ExpirationYearEnd	Two-digit year	2	Card expiration year query value, e.g. 09
TransactionSetupID	GUID	50	Unique GUID that identifies the Transaction Setup ID. The <i>Express</i> Platform generates this.
PASSUpdaterDateTimeBegin		30	Begin date/time of range formatted [yyyy-MM-dd HH:mm:ss.fff]
PASSUpdaterDateTimeEnd		30	End date/time of range formatted [yyyy-MM-dd HH:mm:ss.fff]
PASSUpdaterBatchStatus		2	Specifies whether or not the PASS record will be included in the next PASS Updater batch
■ PASSUpdaterStatus		2	Specifies the match status of the PASS record

Remarks

Used in transactions utilizing our TransForm Tokenization service.

* Response Class

The *Response Class* is used to communicate any output messages back to the sender. An *ExpressResponseCode* is returned with every *Express* method call. This is the first returned variable that should be examined to determine success or failure.

Name	Value	Max Length	Description
Credentials	Any valid Credentials object	N/A	Used to authenticate the sender
Card	Any valid Card object	N/A	Used to pass cardholder information
Batch	Any valid Batch object	N/A	Used to pass batch information
Transaction	Any valid Transaction object	N/A	Used to pass transaction information
ExpressResponseCode	Any valid <i>Express</i> response code	N/A	Express Response Code
ExpressResponseMessage	Any valid <i>Express</i> message	N/A	Express Response Message
HostResponseCode	Any valid host response code	N/A	Host Response Code
HostResponseMessage	Any valid host response message	N/A	Host Response Message
ExpressTransactionDate		N/A	Express platform transaction date
ExpressTransactionTime		N/A	Express platform transaction time in central time
ExpressTransactionTimezone		N/A	Express platform transaction time zone
HostTransactionDateTime		N/A	Host platform transaction date and time
QueryData	Any valid query data	N/A	Query information in xml format
ReportingData	Any valid reporting data	N/A	Transaction information in xml format
ServicesID	Any valid <i>Express</i> services identifier	N/A	Unique services identifier
ReportingID	Any valid Express reporting identifier	N/A	Unique reporting identifier

ScheduledTask Class

The ScheduledTask class is used to schedule transactions.

Name	Value	Max Length	Description
ScheduledTaskID		50	Unique value that identifies the Scheduled Task. The <i>Express</i> Platform generates this.
ScheduledTaskReferenceNumber		50	User generated Reference Number used to identify a Scheduled Task.
Run Frequency	RunFrequency	2	Specifies how often an event should occur.
RunStartDate		8	The start date. Format is YYYYMMDD.
[€] RunUntilCancelFlag	Boolean Type	2	If this is set to true, the task will run every scheduled time until the task is deleted by the user.
RunCycles	Only required when RunUntilCancelFlag is set to false.	4	Number of times this task should run.
■ ScheduledTaskStatus	StatusType	2	The status of the task. You can set this to Active or Disabled.
ScheduledTaskName		50	User generated name of the task.
ScheduledTaskGroupID		50	Express platform generates this.
ScheduledTaskRunLogID		50	Run log ID of a Scheduled Task

ScheduledTaskParameters Class

Field Names

Name	Value	Max Length	Description
ScheduledTaskID		50	Unique GUID that identifies the Scheduled Task. The <i>Express</i> Platform generates this.
ScheduledTaskReferenc eNumber		50	User generated Reference Number used to identify a Scheduled Task.
Scheduled Task Query Type		2	Query type used to provide additional event information (0=Default, 1=RunLog)
RunStartDate		8	The start date. Format is YYYYMMDD.
■ RunUntilCancelFlag	ExtendedBoolean Type	2	If this is set to true, the task will run every scheduled time until the task is deleted by the user.
₽ RunFrequency	ExtendedRunFrequency	2	Specifies how often an event should occur.
■ Scheduled Task Status	ExtendedStatusType	2	The status of the task. You can set this to Active or Disabled.
■ ScheduledTaskRunStatus		2	The run status of the task.
ScheduledTaskName		50	User generated name of the task.
ScheduledTaskGroupID		50	Express platform generates this.

Remarks

Used in transactions utilizing our Scheduled Task service.

Terminal Class

Name	Value	Max Length	Description
▼ TerminalID	Any valid value: Example 01	40	Unique Terminal Identifier assigned by Element
<u>TerminalType</u>	Any valid TerminalType enumeration	2	Type of terminal
TerminalCapabilityCode	Any valid TerminalCapabilityCode enumeration	2	Terminal Capability Code
<u>TerminalEnvironmentCode</u>	Any valid TerminalEnvironmentCode enumeration	2	Terminal Environment Code
₽ CardPresentCode	Any valid CardPresentCode enumeration	2	Card Present Code
E CVVPresenceCode	Any valid CVVPresenceCode enumeration	2	Code that indicates the status of CVV
EP CVVResponseType	Any valid CVVResponseType enumeration	2	Card Verification Value Response Type
<u>CardInputCode</u>	Any valid CardInputCode enumeration	2	Card Input Code
CardholderPresentCode	Any valid CardholderPresentCode enumeration	2	Cardholder Present Code
<u>MotoECICode</u>	Any valid MotoECICode enumeration	2	Mail Order Telephone Order Electronic Commerce Indicator Code
E ConsentCode	Any valid ConsentCode enumeration	2	Consumer Consent Code
TerminalEncryptionFormat	Any valid TerminalEncryptionFormat enumeration	2	Not used
₱ TerminalSerialNumber	Any valid TerminalSerialNumber	40	Terminal Serial Number
LaneNumber	Any valid LaneNumber	6	Lane Number

Transaction Class

Name	Value	Max Length	Description
▼ TransactionID		10	Unique transaction identifier
TransactionAmount		10	Dollar amount of transaction (Paymentech <= 99999.99)
Original Authorized Amount		10	Original dollar amount authorized
SalesTaxAmount		12	Dollar sales tax amount
TipAmount		12	Dollar tip amount
CashBackAmount		12	Cash back amount
ApprovalNumber		30	Issuer assigned approval number
ClerkNumber		10	Clerk number
ShiftID		10	Shift ID
ReferenceNumber		50	User defined reference number
■PReversalType		2	Specifies the type of reversal transaction
 ■ MarketCode		2	The Market type of the merchant
AcquirerData		1000	Refer to host.
■ BillPaymentFlag		2	Specifies if this is a Bill Payment transaction
[©] DuplicateCheckDisableFlag	Boolean Type	2	Disables duplicate checking for this transaction
[■] DuplicateOverrideFlag	Boolean Type	2	Overrides duplicate transaction checking at the host for transactions that were previously returned as being a duplicate
₽ RecurringFlag	Boolean Type	2	Specifies if this is part of a recurring series of transactions
▼ TicketNumber		50	Used for Direct Marketing, MOTO, and E-Commerce transactions. Required when card number is manually keyed.
CommercialCardCustom		25	Commercial card customer code

Name	Value	Max Length	Description
erCode			
TransactionStatusCode		2	Status/State of transaction
TransactionStatus		25	Description of status/state of transaction
TransactionSetupID	GUID	50	Unique GUID that identifies the Transaction Setup ID. The <i>Express</i> Platform generates this.
ApprovedAmount		10	Approved amount
ConvenienceFeeAmount		10	Convenience fee amount (Vantiv only)
『 PartialApprovedFlag	Boolean Type	1	Indicates whether software supports partial approvals
MerchantVerificationValue		10	Merchant Verification Value (TSYS, FDC, or Global only). Can be included or left out and picked up on auth network profile.
© CommercialCardResponseCode		2	Commercial card response code (TSYS or FDC or Vantiv only)
		13	Balance amount. If negative, a minus sign will precede the amount. Format: DDDDDDDDD.CC
₱ BalanceCurrencyCode		3	Balance amount currency code. Numeric. 840 = US dollars 124 = Canadian dollars 978 = Euro
BillPayerAccountNumber		25	Bill payer account number

** Token Class

Field Names

Name	Value	Max Length	Description
₱ TokenID	Valid Token ID		Unique Token ID representing the card
▼ TokenProvider	Any valid TokenProvider enumeration		TokenProvider enumeration Null = 0 ExpressPASS = 1 OmniToken = 2 Paymetric = 3 TransArmor = 4
	Valid Merchant ID		Merchant Vault ID
■ TokenProviderID	TransArmor Provider ID	3	TransArmor Provider ID

**TransactionSetup Class

Name	Value	Max Length	Description
Transaction Setup ID	GUID	50	Unique GUID that identifies the Transaction Setup ID. The <i>Express</i> Platform generates this.
<u>Interpolation Setup Method</u>	Any valid TransactionSetupMethod enumeration	2	Transaction method used
■ P <u>Device</u>	Any valid Device enumeration	2	Entry device used
■ Embedded	Boolean Type	2	Specifies if card entry page will be embedded in an application
 ©CVVRequired	Boolean Type	2	Requires user to enter CVV2, CVC2, or CID values
₽ PAutoReturn	Boolean Type	2	Specifies if approval page will be automatically directed to ReturnURL
CompanyName		50	Name of merchant
		150	HTTPS URL of merchant logo to be displayed
▼ Tagline		150	Text displayed next to logo
		250	Text to display to customer (if not provided, generic message will be displayed)
ReturnURL		150	Return URL link displayed after transaction completion or

Name	Value	Max Length	Description
			cancelled
ReturnURLTitle		100	Text to display for ReturnURL (if not provided generic "return to merchant" message will be displayed
OrderDetails		250	Text to describe purchase or detail information
ProcessTransactionTitle		100	Text to describe submit button
		50	Unique code returned from a TransactionSetup and as part of the query string redirect during a successful Hosted Payments request.
□ PDeviceInputCode	Any valid DeviceInputCode enumeration	2	Device input code

Enumerations

CardholderPresentCode

CardholderPresentCode specifies the location of the cardholder at the time of the transaction.

Enumeration Names

Name	Value	Description
■ Default	0	Default
 Unknown	1	Unknown
■ Present	2	Cardholder is present
■ NotPresent	3	Cardholder is not present
[ॗ] MailOrder	4	Mail order
PhoneOrder	5	Phone order
StandingAuth	6	Standing Auth
ECommerce	7	ECommerce

CardInputCode

CardInputCode specifies the means in which the Card Number or Track Data was acquired.

Enumeration Names

Name	Value	Description
■ Default	0	Default
F Unknown	1	Unknown
MagstripeRead	2	Magstripe Read
ContactlessMagstripeRead	3	Contactless Magstripe Read
ManualKeyed	4	Manual Keyed
ManualKeyedMagstripeFailure	5	Manual Keyed Magstripe Failure

CheckType

CheckType specifies the type of check being used in the transaction.

Enumeration Names

Name	Value	Description
Personal	0	Personal
Business	1	Business

ConsentCode

ConsentCode specifies the way the consumers consent for the transaction was received.

Enumeration Names

Name	Value	Description
■ NotSpecified	0	NotSpecified
■ FaceToFace	1	FaceToFace
[©] Phone	2	Phone
[™] Internet	3	Internet

CVVResponseType

CVVResponseType specifies the type of CVV Response codes that are returned by the host. The Extended value only applies to Visa cards.

Enumeration Names

Name	Value	Description
■ Regular	0	Regular
Extended	1	Extended

DDAAccountType

DDAAccountType specifies the type of DDA Account being used in the transaction.

Enumeration Names

Name	Value	Description
f Checking	0	Checking
[□] Savings	1	Savings

ReversalType

ReversalType specifies the type reversal that is being performed. See the description of reversals in the beginning of this document for more information.

Enumeration Names

Name	Value	Description
	0	System
Full	1	Full
	2	Partial

BooleanType

BooleanType is a true or false value used for setting various flags in the system.

Enumeration Names

Name	Value	Description
False	0	False
□ True	1	True

CardPresentCode

CardPresentCode specifies the location of the card at the time of the transaction.

Enumeration Names

Name	Value	Description
[□] Default	0	Default
ு Unknown	1	Unknown
Present	2	Present
■ NotPresent	3	Not Present

CVVPresenceCode

CVVPresenceCode specifies the status of the CVV code from the consumer card as it pertains to the transaction. If the CVV code from the card is not included in the transaction, it is recommended that this code be set to the reason why.

Enumeration Names

Name	Value	Description
[©] UseDefault	0	Use Default
Not Provided	1	Not Provided
₽ Provided	2	Provided
ा lllegible	3	Illegible
Customer Illegible	4	Customer Illegible

BatchCloseType

BatchCloseType specifies the process you want to use when closing your batch. See the BatchClose method for more information.

Enumeration Names

Name	Value	Description
[™] Regular	0	Regular
☞ Force	1	Force

BatchIndexCode

BatchIndexCode specifies which Batch you are querying against.

Enumeration Names

Name	Value	Description
□ Current	0	Current open batch
FirstPrevious	1	1 st Previous batch

■ BatchGroupingCode

This enumeration is reserved for future use.

Enumeration Names

Name	Value	Description
ਾਂ FullBatch	0	FullBatch
[ॗ] SingleBatch	1	SingleBatch

BatchQueryType

This enumeration is reserved for future use.

Name	Value	Description
ः Total	0	Total
ः Item	1	Item

MarketCode

MarketCode specifies the industry type of the merchant. Set this to a value of *Default* if you would like to use the *MarketCode* from the merchant profile.

Enumeration Names

Name	Value	Description
o □ Default	0	Default
AutoRental	1	AutoRental
DirectMarketing	2	DirectMarketing
■ ECommerce	3	ECommerce
FoodRestaurant	4	FoodRestaurant
F HotelLodging	5	HotelLodging
■ Petroleum	6	Petroleum
■ Retail	7	Retail
o QSR	8	QSR

MotoECICode

MotoECI Code is used on MOTO and E-Commerce transactions to identify the type of transaction, and the means in which it was obtained.

Name	Value	Description
^ॼ Default	0	Default
□ NotUsed	1	NotUsed
[©] Single	2	Single
[©] Recurring	3	Recurring
Installment	4	Installment
SecureElectronicCommerce	5	Secure Electronic Commerce
NonAuthenticatedSecureTransaction	6	NonAuthenticated Secure Transaction
NonAuthenticatedSecureECommerceT ransaction	7	NonAuthenticated Secure eCommerce Transaction. Use for all <i>Express</i> eCommerce transactions.
NonSecureECommerceTransaction	8	NonSecure eCommerce Transaction

PaymentAccountType

PaymentAccountType indicates the type of Payment Account that is being created.

Enumeration Names

Name	Value	Description
	0	CreditCard
F Checking	1	Checking
	2	Savings
[⊕] ACH	3	ACH
 Other	4	Other

RunFrequency

RunFrequency specifies the how often an event should occur.

Enumeration Names

Name	Value	Description
	0	OneTimeFuture
□ Daily	1	Daily
[™] Weekly	2	Weekly
≅ BiWeekly	3	BiWeekly
[™] Monthly	4	Monthly
■ BiMonthly	5	BiMonthly
	6	Quarterly
F SemiAnnually	7	SemiAnnually
[©] Yearly	8	Yearly

■ TerminalCapabilityCode

TerminalCapabilityCode specifies what the capabilities of the Terminal are. For example: an eCommerce website would be considered *KeyEntered*.

Name	Value	Description
■ Default	0	Default
 Unknown	1	Unknown
■ NoTerminal	2	No Terminal
	3	Magstripe Reader
ContactlessMagstripeReader	4	Contactless Magstripe Reader
F KeyEntered	5	Key Entered

TerminalEnvironmentCode

TerminalEnvironmentCode specifies what type of conditions the Terminal is operated in. For example: a card reader at a fuel pump would be considered LocalUnattended.

Enumeration Names

Name	Value	Description
o ■ Default	0	Default
■ NoTerminal	1	No Terminal
■ LocalAttended	2	Local Attended
□ LocalUnattended	3	Local Unattended
■ RemoteAttended	4	Remote Attended
■ RemoteUnattended	5	Remote Unattended
■ ECommerce	6	ECommerce

TerminalType

TerminalType specifies what type of Terminal is used to send the transaction in.

Name	Value	Description
□ Unknown	0	Unknown
■ PointOfSale	1	PointOfSale
■ ECommerce	2	eCommerce
 мото	3	мото
[©] FuelPump	4	FuelPump
₽ ATM	5	ATM
[□] Voice	6	Voice

TransactionStatusCode

TransactionStatusCode identifies the state/status the transaction is in. It provides a means for applications to know the state of a transaction without having to store it locally.

Enumeration Names

Name	Value	Description
inProcess	0	InProcess
□ Approved	1	Approved
of Declined □	2	Declined
□ Duplicate	3	Duplicate
₽ Voided	4	Voided
Authorized	5	Authorized
■ AuthCompleted	6	AuthCompleted
Reversed	7	Reversed
[■] Success	8	Success
Returned	9	Returned
■ Pending	10	Pending
Queued	11	Queued
■ Unknown	12	Unknown
Error	13	Error
□ Originated	14	Originated
■ Settled	15	Settled
	16	PartialApproved
[□] Rejected	17	Rejected

StatusType

StatusType specifies current status of an item.

Name	Value	Description
■ Active	0	Active
■ Disabled	1	Disabled
☞ Removed	2	Removed

ExtendedBooleanType

ExtendedBooleanType is a true, false, or null value used for setting various flags in the system.

Enumeration Names

Name	Value	Description
[©] Null	-1	Null
False	0	False
ः True	1	True

ExtendedStatusType

ExtendedStatusType specifies current status of an item.

Enumeration Names

Name	Value	Description
[□] Null	-1	Null
₫ Active	0	Active
[■] Disabled	1	Disabled
■ Removed	2	Removed

ExtendedRunFrequency

ExtendedRunFrequency specifies the how often an event should occur.

Name	Value	Description
o P Null	-1	Null
OneTimeFuture	0	OneTimeFuture
☞ Daily	1	Daily
□ F Weekly	2	Weekly
■ BiWeekly	3	BiWeekly
	4	Monthly
■ BiMonthly	5	BiMonthly
■ Quarterly	6	Quarterly
	7	SemiAnnually
[©] Yearly	8	Yearly

ScheduledTaskRunStatus

ScheduledTaskRunStatus specifies the current run status of a task.

Enumeration Names

Name	Value	Description
₽ Null	-1	Null
₽ Pending	0	Pending
F Running	1	Running
	2	Completed
₽ Error	3	Error
■ SystemRetry	4	SystemRetry
	5	ManualRetry

Device

Device specifies the entry device used for the transaction.

Enumeration Names

Name	Value	Description
■ Default	0	Default
MagtekEncryptedSwipe	1	MagtekEncryptedSwipe
EncryptedInputDevice	2	EncryptedInputDevice
(Reserved for future use)		

TransactionSetupMethod

TransactionSetupMethod specifies the transaction method used for the transaction.

Name	Value	Description
[ॗ] Default	0	Default
☐ CreditCardSale	1	CreditCardSale
CreditCardAuthorization	2	CreditCardAuthorization
CreditCardAVSOnly	3	CreditCardAVSOnly
□ CreditCardForce	4	CreditCardForce
□ DebitCardSale	5	DebitCardSale (Element recommends using Sale (value = 9) instead)
CheckSale (Reserved for future use)	6	CheckSale
PaymentAccountCreate	7	PaymentAccountCreate
	8	PaymentAccountUpdate
[ॗ] Sale	9	Sale

Healthcare[First][Second][Third][Fourth]AccountType

Healthcare Account Type specifies the type of healthcare account.

Enumeration Names

Name	Value	Description
■ NotSpecified	0	Not Specified
Savings	1	Savings
Checking	2	Checking
☐ CreditCard	3	Credit Card
[©] Universal	4	Universal
➡ StoredValueAccount	5	Stored Value Account
CashBenefitsAccount	6	Cash Benefits Account
FoodStampsAccount	7	Food Stamps Account

Healthcare[First][Second][Third][Fourth]AmountType

Healthcare Amount Type specifies the type of healthcare amount.

Name	Value	Description
■ LedgerBalance	0	Ledger Balance
AvailableBalance	1	Available Balance
⊟ealthcare	2	Healthcare
ਾਂ Transit	3	Transit
[©] Copayment	4	Copayment
ு OriginalAmount	5	Original Amount
Partial Authorized Amount	6	Partial Authorized Amount
Prescription	7	Prescription
	8	Vision
F Clinic	9	Clinic
[©] Dental	10	Dental
[©] CashOver	11	Cash Over
OriginalCashOver	12	Original Cash Over

Healthcare[First][Second][Third][Fourth]AmountSign

Healthcare Amount Sign specifies the healthcare amount sign.

Enumeration Names

Name	Value	Description
Positive	0	Positive
■ Negative	1	Negative

LodgingSpecialProgramCode

LodgingSpecialProgramCode specifies whether a cardholder is participating in a special program.

Enumeration Names

Name	Value	Description
□ □ Default □	0	Default
Sale	1	Sale. When using this value, LodgingChargeType must be 1/Restaurant or 2/GiftShop.
■ NoShow	2	No Show
AdvanceDeposit	3	Advance Deposit

LodgingChargeType

LodgingChargeType specifies a hotel charge type.

Enumeration Names

Name	Value	Description
■ Default	0	Default
■ Restaurant	1	Restaurant
GiftShop	2	Gift Shop

LodgingPrestigiousPropertyCode

LodgingPrestigiousPropertyCode specifies whether a hotel is participating in Visa's prestigious property program.

NonParticipant	0	Non-Participant
☐ DollarLimit500	1	Prestigious Property with \$500 Limit
■ DollarLimit1000	2	Prestigious Property with \$1000 Limit
DollarLimit1500	3	Prestigious Property with \$1500 Limit

AutoRentalVehicleClassCode

AutoRentalVehicleClassCode specifies the classification of rental vehicle.

Name	Value	Description
[™] Unused	0	Unused
o∓ Mini	1	Mini
	2	Subcompact
Economy	3	Economy
[©] Compact	4	Compact
F Midsize	5	Midsize
■ Intermediate	6	Intermediate
	7	Standard
Fullsize	8	Fullsize
₽ Luxury	9	Luxury
■ Premium	10	Premium
MiniVan	11	Mini Van
☑ Van12Passenger	12	12-Passenger Van
MovingVan	13	Moving Van
■ Van15Passenger	14	15-Passenger Van
☐ CargoVan	15	Cargo Van
☐ Truck12Foot	16	12-Foot Truck
☐ Truck20Foot	17	20-Foot Truck
☐ Truck24Foot	18	24-Foot Truck
☐ Truck26Foot	19	26-Foot Truck
[™] Moped	20	Moped
Stretch	21	Stretch
■ Regular	22	Regular
ு Unique	23	Unique
[©] Exotic	24	Exotic
☐ TruckSmallMedium	25	Small or Medium Truck
☐ Truck Large	26	Large Truck
■ SUVSmall	27	Small SUV
[■] SUVMedium	28	Medium SUV
■ SUVLarge	29	Large SUV
SUVExotic	30	Exotic SUV
FourWheelDrive	31	Four-Wheel Drive
Special	32	Special
Miscellaneous	33	Miscellaneous

AutoRentalDistanceUnit

AutoRentalDistanceUnit specifies the distance unit of measurement.

Enumeration Names

Name	Value	Description
Unused	0	Unused
Miles	1	Miles
F Kilometers	2	Kilometers

AutoRentalAuditAdjustmentCode

AutoRentalAuditAdjustmentCode specifies if an adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement.

Enumeration Names

Name	Value	Description
[©] NoAdjustments	0	No Adjustments
MultipleAdjustments	1	Multiple Adjustments
OneAdjustmentCardmemberNotified	2	One Adjustment Cardmember Notified
₽	3	One Adjustment Cardmember Not Notified
OneAdjustmentCardmemberNotNotified		

■ PASSUpdaterBatchStatus

PASSUpdaterBatchStatus specifies whether or not the TransForm Tokenization record will be included in the next TransForm Tokenization Updater batch. Used with TransForm Tokenization Auto Updater.

Enumeration Names

Name	Value	Description
■ Null	0	Null
IncludedInNextBatch	1	Included in next batch
NotincludedInNextBatch	2	Not included in next batch

₽ PASSUpdaterOption

PASSUpdaterOption specifies whether or not the TransForm Tokenization record will be updated automatically. Used with TransForm Tokenization Auto Updater.

Name	Value	Description
■ Null	0	Null
	1	Auto update enabled
AutoUpdateDisabled	2	Auto update disabled

PASSUpdaterStatus

PASSUpdaterStatus specifies the match status of the TransForm Tokenization record. Used with TransForm Tokenization Auto Updater.

Enumeration Names

Name	Value	Description
o P Null	0	Null
UpdateInProgress	1	Update in progress
MatchNoChanges	2	Match, no changes
MatchAccountChange	3	Match, account change
MatchExpirationChange	4	Match, expiration change
MatchAccountClosed	5	Match, account closed
MatchContactCardholder	6	Match, contact cardholder
NoMatchParticipating	7	No match, participating
NoMatchNonParticipating	8	No match, non-participating
	9	Invalid info
NoResponse	10	No response
■ NotAllowed	11	Not allowed
 Error	12	Error
PASSUpdaterDisabled	13	PASS Updater disabled
■ NotUpdated	14	Not updated

DeviceInputCode

DeviceInputCode specifies the entry device input type used for the transaction.

Name	Value	Description
■ NotUsed	0	Not Used
■ Unknown	1	Unknown
■ Terminal	2	Terminal
≅ Keyboard	3	Keyboard

TerminalEncryptionFormat

TerminalEncryptionFormat specifies the encryption format of the device used for the transaction. This field is not currently used and should be left as Default/0.

Enumeration Names

Name	Value	Description
□ Default	0	Default
Format1	1	Format 1
Format2	2	Format 2
Format3	3	Format 3
Format4	4	Format 4

EncryptedFormat

EncryptedFormat specifies the encryption format of the device used for the transaction.

Enumeration Names

Name	Value	Description
□ Default	0	Default
Format1	1	Magtek
■Format2	2	Ingenico DPP
Format3	3	Ingenico On-Guard
Format4	4	ID Tech
Format5	5	VeriFone Vx Devices w/Mercury Format
Format6	6	Infinite Peripherals
Format7	7	VeriFone Mx Devices

TokenProvider

TokenProvider specifies the provider of the tokenization solutions.

Name	Value	Description
[™] Null	0	Null
■ ExpressPASS	1	ExpressPASS
omniToken	2	OmniToken
Paymetric	3	Paymetric
■ TransArmor	4	TransArmor

Appendix

Express Response Codes

An *ExpressResponseCode* is returned with every *Express* method call. This is the first returned variable that should be examined to determine success or failure. An *ExpressResponseCode* = 0 indicates an approval or successful request. An *ExpressResponseCode* = 5 indicates a partial approval request. An *ExpressResponseCode* <> 0 or 5 indicates a decline or non-approval.

Code	Description
0	Approved / Success
5	Partial Approval
20	Decline
21	Expired card
22	Duplicate approved
23	Duplicate
24	Pick up card
25	Referral / Call Issuer
30	Balance Not Available
90	Not defined
101	Invalid data
102	Invalid account
103	Invalid request
104	Authorization failed
105	Not Allowed
120	Out of Balance
1001	Communication error
1002	Host error
1009	Error

AVS Response Codes

Code	Description		
Α	Address: Address matches, zip does not match. For Discover, when using TSYS or Paymentech platforms, both address and zip code match. For Discover, when using Global, First Data, or Vantiv platforms, only address matches.		
В	Incompatible formats (postal code): Street addresses match. Postal code not verified due to incompatible formats (Acquirer sent both street address and postal code)		
С	Incompatible formats (all information): Street address and postal code not verified due to incompatible formats. (Acquirer sent both street address and postal code.		
D	Street addresses and postal codes match		
E	Edit error: For example, AVS not allowed for this transaction		
F	International Transaction: Street address and postal code match		
G	Global non-AVS participant.		
1	International Transaction: Address information not verified for international transaction.		
J	American Express only. Card Member Information and Ship-To information Verified – Fraud Protection Program.		
K	American Express only. Card Member Information and Ship-To Information Verified – Standard.		
M	Match: Street addresses and postal codes match.		
N	No: Address and zip code do not match		
Р	Postal codes match. Street address not verified due to incompatible formats (Acquirer sent both street address and postal code)		
R	Retry: System unavailable or timed out		
S	Service not Supported: Issuer does not support AVS and Visa, INAS, or the issuer processing center		
Т	Nine-digit zip code matches, address does not match		
U	Unavailable: Address information not verified for domestic transactions		
w	Whole zip: Nine-digit zip code matches, address does not match. For Discover, no data provided.		
X	Exact: Address and nine-digit zip code match		
Y	Yes: Address and five-digit zip code match. For Discover, when using TSYS or Paymentech platforms, only address matches. For Discover, when using Global, First Data, or Vantiv platforms, both the address and five-digit zip code match.		
Z	Zip: Five-digit zip code matches, address does not match		
0	No address verification has been requested (TSYS only).		

CVV Response Codes

Code	Description	
М	Match	
N	No Match	
Р	Not Processed	
S	CVV value should be on the card, but the merchant has indicated that it is not present (Visa & Discover only)	
U	Issuer is not certified for CVV processing	
X	Service provider did not respond	

CAVV Response Codes

Code	Description		
Space	CAVV not present		
0	CAVV authentication results invalid		
1	CAVV failed validation – authentication		
2	CAVV passed validation – authentication		
3	CAVV passed validation – attempt		
4	CAVV failed validation – attempt		
6	CAVV not validated, issuer not participating in CAVV validation		
7	CAVV failed validation – attempt (US issued card, non-US merchant)		
8	CAVV passed validation – attempt (US issued card, non-US merchant)		
9	CAVV failed validation – attempt (US issued card, non-US merchant, issuer authentication server not available)		
Α	CAVV passed validation – attempt (US issued card, non-US merchant, issuer authentication server not available)		
В	CAVV passed validation – information only, no liability shift		

Commercial Card Response Codes

The *CommercialCardResponseCode* value may returned for Visa and MasterCard commercial cards on the TSYS or FDC or Vantiv platform only.

Code	Description		
В	Business Card		
D	Visa Commerce (reserved for future use)		
R	Corporate Card		
S	Purchasing Card		
Т	Unable to obtain information		
0	Non-commercial Card		
Space	Non-commercial Card or request error		

Sample Code

HealthCheck

```
[C#] Web Service
//Declare variables
ElementExpress.Express express = new ElementExpress.Express ();
ElementExpress.Credentials credentials = new ElementExpress.Credentials ();
ElementExpress.Application application = new ElementExpress.Application ();
ElementExpress.Response response = new ElementExpress.Response ();
ElementExpress.Transaction transaction = new ElementExpress.Transaction ();
//Setup the Application and Credential objects
application.ApplicationID = "Inset Application ID Here";
application.ApplicationName = "Insert Application Name Here";
application.ApplicationVersion = "Insert Application Version Here";
credentials.AccountID = "Insert AccountID Here";
credentials.AccountToken = "Insert AccountToken Here";
credentials.AcceptorID = "Insert MerchantID Here";
//Output the returned values to the Response object
response = express.HealthCheck (credentials, application);
MessageBox.Show ("ExpressMessage= "+ response.ExpressResponseMessage+ "\n HostMessage= "+
response.HostResponseMessage);
```

```
[C#] XML Post
//Setup the XML request
System.Net.WebRequest request = System.Net.WebRequest.Create
("https://certtransaction.elementexpress.com/");
request.Method = "POST";
request.ContentType = "text/xml";
System.IO.StreamWriter writer = new System.IO.StreamWriter (request.GetRequestStream ());
writer.Write ("<HealthCheck xmlns='https://transaction.elementexpress.com'>");
writer.Write ("<Credentials>");
writer.Write ("<AccountID>Insert Account ID Here</AccountID>");
writer.Write ("<AccountToken>Insert Acct Token Here</AccountToken>");
writer.Write ("<AcceptorID>Insert Acceptor ID Here</AcceptorID>");
writer.Write ("</Credentials>");
writer.Write ("<Application>");
writer.Write ("<ApplicationID>Insert App ID Here</ApplicationID>");
writer.Write ("<ApplicationName>Insert App Name Here</ApplicationName>");
writer.Write ("<ApplicationVersion>Insert App Version Here</ApplicationVersion>");
writer.Write ("</Application>");
writer.Write ("</HealthCheck>");
writer.Close ();
//Output the XML response
System.Net.WebResponse response = request.GetResponse ();
System.IO.StreamReader reader = new System.IO.StreamReader (response.GetResponseStream ());
string sXml = reader.ReadToEnd().Trim();
MessageBox.Show (sXml);
request.GetRequestStream ().Close ();
response.GetResponseStream ().Close ();
```

```
[VB6] XML Post
Dim oXmlHttp As New MSXML2.XMLHTTP
Dim sXml As String
sXml = sXml & "<HealthCheck>"
sXml = sXml & "<Credentials>"
sXml = sXml & "<AccountID>Insert AccountID here</AccountID>"
sXm1 = sXm1 & "<AccountToken>Insert AccountToken here/AccountToken>"
sXml = sXml & "<AcceptorID>Insert AcceptorID here</AcceptorID>"
sXml = sXml & "</Credentials>"
sXml = sXml & "<Application>"
sXml = sXml & "<ApplicationID>Insert ApplicationID here/ApplicationID>"
sXml = sXml & "<ApplicationName>Insert ApplicationName here/ApplicationName>"
sXml = sXml & "<ApplicationVersion>Insert ApplicationVersion here</ApplicationVersion>"
sXml = sXml & "</Application>"
sXml = sXml & "</HealthCheck>"
oXmlHttp.open "POST", "https://certtransaction.elementexpress.com", False
oXmlHttp.setRequestHeader "Content-Type", "text/xml"
oXmlHttp.send sXml
MsgBox oXmlHttp.responseText
Set oXmlHttp = Nothing
```

CreditCardSale

```
[C#] Web Service
//Declare variables
ElementExpress.Express express = new ElementExpress.Express ();
ElementExpress.Application application = new ElementExpress.Application ();
ElementExpress.Credentials credentials = new ElementExpress.Credentials ();
ElementExpress.Card card = new ElementExpress.Card ();
ElementExpress.Terminal terminal = new ElementExpress.Terminal ();
ElementExpress.Transaction transaction = new ElementExpress.Transaction ();
ElementExpress.Response response = new ElementExpress.Response ();
//Setup the Application, Credential, Terminal
//Transaction, and Card objects
application.ApplicationID = "Inset Application ID Here";
application.ApplicationName = "Insert Application Name Here";
application.ApplicationVersion = "Insert Application Version Here";
credentials.AccountID = "Insert AccountID Here";
credentials.AccountToken = "Insert AccountToken Here";
credentials.AcceptorID = "Insert MerchantID Here";
terminal.TerminalID = "01";
terminal.CardholderPresentCode = ElementExpress.CardholderPresentCode.Present;
terminal.CardInputCode = ElementExpress.CardInputCode.ManualKeyedMagstripeFailure;
terminal.TerminalCapabilityCode = ElementExpress.TerminalCapabilityCode.MagstripeReader;
terminal.TerminalEnvironmentCode = ElementExpress.TerminalEnvironmentCode.LocalAttended;
terminal.CardPresentCode = ElementExpress.CardPresentCode.Present;
terminal.MotoECICode = ElementExpress.MotoECICode.NotUsed;
transaction.MarketCode = ElementExpress.MarketCode.Retail;
transaction.TransactionAmount = "6.54";
card.CardNumber = "5499990123456781";
card.ExpirationMonth = "09";
card.ExpirationYear = "08";
//Output returned values to the Response object
```

response = express.CreditCardSale(credentials, application, terminal, card, transaction, null,
null);

MessageBox.Show ("ExpressMessage=" + response. ExpressResponseMessage + "\n HostMessage=" +
response. HostResponseMessage);

NULL Processor

The NULL Processor is a host simulator that provides a means for integrators to develop and test their applications by passing in certain parameters to receive expected results.

The below table works with the following transaction types: *CreditCardSale, CreditCardAuthorization, CreditCardCredit, CreditCardReturn, DebitCardSale and DebitCardReturn.*

Dollar Amount	Express Response Code/Message	Host Response Code/Message
\$1.00	0 / APPROVED	000 / AP
\$23.05	5 / PARTIAL APPROVED	010 / PARTIAL AP
\$.20	20 / DECLINED	007 / DECLINED
\$.21	21 / EXPIRED CARD	054 / EXPIRED CARD
\$.22	22 / DUPLICATE AP	094 / AP DUP
\$.23	23 / DUPLICATE	094 / DUPLICATE
\$.24	24 / PICK UP CARD	004 / PICK UP CARD
\$.25	25 / CALL ISSUER	002 / CALL ND
\$.90	90 / UNDEFINED	
\$1.01	101 / INVALID DATA	
\$1.02	102 / INVALID ACCOUNT	
\$1.03	103 / INVALID REQUEST	
\$1.04	104 / AUTH FAILED	
\$1.05	105 / NOT ALLOWED	058 / UNAUTH TRANS
\$1.20	120 / OUT OF BALANCE ONB / INV BAL/SETTL	
\$10.01	1001 / COMM ERROR	
\$10.02	1002 / HOST ERROR	
\$10.09	1009 / ERROR	
\$23.06	0 / APPROVED	000 / AP (Balance and Currency returned)
Any other amount	0 / APPROVED	000 / AP

The below table works with the following transaction types: *CheckSale*. This provides integrators the ability to view CheckSale transactions in various status values they may encounter.

Dollar Amount	Express Response Code/Message	Transaction Status Code / Status
\$33.33	0 / Success	9 / Returned
\$33.34	0 / Success	10 / Pending
\$33.35	0 / Success	11 / Queued
\$33.36	0 / Success	12 / Unknown
\$33.37	0 / Success	13 / Error
\$33.38	0 / Success	14 / Originated
\$33.39	0 / Success	15 / Settled

AVS Response Values

Billing Address 1	Billing Zipcode	AVS Response Code / Meaning	
4	Any Other Zipcode	A / Address Match	
4	30329	M / Address and Zipcode Match	
Any Other Address	Any Other Zipcode	N / No Match	
4	30330	Y / Address and Zipcode Match	
Any Other Address	30330	Z / Zipcode Match	

CVV Response Values

CVV Value	CVV Response Code / Meaning	
123	M / Match	
321	N / No Match	
333	P / Not Processed	

Terminal Settings

For a list of suggested Terminal settings, please contact the Express Certification team.

Interface URL's

Interface	Platform	Туре	URL
Transaction	Certify	XML	https://certtransaction.elementexpress.com/
Transaction	Certify	SOAP	https://certtransaction.elementexpress.com/express.asmx
Reporting	Certify	XML	https://certreporting.elementexpress.com/
Reporting	Certify	SOAP	https://certreporting.elementexpress.com/express.asmx
Services	Certify	XML	https://certservices.elementexpress.com/
Services	Certify	SOAP	https://certservices.elementexpress.com/express.asmx

Express Certification Team

To contact the Express Certification team, please email certification@vantiv.com