worldpay
from FIS

Express Interface Specification

Version 3.1.2

August 2023

Worldpay Express™ Interface Specification
Disclaimer, Terms of Use and Compliance Representations

Disclaimer

This **Interface Specification** and all documentation contained herein or provided to you hereunder (the "Specifications") are licensed by Worldpay Integrated Payments, LLC ("WIP"), ("Licensor") on an "AS IS" basis. No representations or warranties are expressed or implied, including, but not limited to, warranties of suitability, quality, merchantability, or fitness for a particular purpose (irrespective of any course of dealing, custom or usage of trade), and all such warranties are expressly and specifically disclaimed. Licensor shall have no liability or responsibility to you or any other person or entity with respect to any liability, loss, or damage, including lost profits whether foreseeable or not, or other obligation for any cause whatsoever, caused or alleged to be caused directly or indirectly by the Specifications. Use of the Specifications signifies agreement with the disclaimer set forth in this paragraph and the below license and restricted use terms and conditions.

Ownership and Restricted Terms of Use

Ownership of all Specifications, related documentation and all intellectual property rights therein and thereto shall remain at all times with Licensor. All rights not expressly granted to you herein are reserved to Licensor. Licensor grant you the right to use the Specification for the sole purpose of transmitting transactions directly to WIP from a merchant transaction originating device. Under no circumstances may you reverse engineer, translate, re-direct, emulate, disseminate to other entities, decompile, adapt, or disassemble the information contained in the Specifications nor shall you attempt to create source code/object code to emulate the Express platform. You agree that the Specifications and the printed materials and documentation that accompany these Specifications are the confidential information of Licensor and may not be used except as otherwise expressly permitted herein.

Compliance Representations

You represent, warrant and agree:

- To comply with the card brand operating regulations and all applicable PCI Data Security Standards ("PCI DSS") and Payment Application Data Security Standards ("PA-DSS") and all requirements applicable to the distribution and use of these Specifications, Worldpay IP products, and your payment application.

- To comply with all applicable federal, state and local laws, rules and regulations, including those related or pertaining to privacy and truncating and masking cardholder account information and data.

- To follow and abide by this specification and any modifications made to same from time to time by Licensors. The most recent version of this specification can be obtained from a WIP Developer Integrations Consultant.

- To distribute and otherwise make available integration upgrades or modifications made by WIP to your reseller and merchant base utilizing any WIP product integrated by you or your payment application.

Any use of your point of sale system or environment to store, process or transmit cardholder data, places that point of sale system and environment within full scope of the PA-DSS. You agree to comply with PCI DSS and, if your point of sale system or environment engages in any of the above activities, to ensure that the point of sale system and environment meets all PA-DSS requirements.

 By your use of these Specifications, you warrant, represent and certify your agreement to the above terms and conditions and that your payment application is compliant and adheres to the above requirements and that future versions of your payment application will continue to comply and adhere with these requirements.

Document History

Date	Name	Change Description	Version
2023-08-30	Jeff Gross	• Clarified TransactionID use within EnhancedBINQuery method. • Updated AVS Response Codes table indicating that AVS RC values of R and Y may also be returned on International transactions per Visa. • Clarified max length of FraudSightRiskStatus since both numeric or text responses may be returned. • Added DDAAccountType as conditional input parameter in CheckQuery method. • Added note under AVS Response Code table about AVS value that Visa may no longer return starting in April 2023. • Added RCode and BOE as possible response parameters in CheckQuery method, and added them to Transaction class. • Added Secure3DSVersion and Secure3DSDirectoryServerTranID to Transaction Class in support of MasterCard SLI 3DS requirements. • Added clarifying notes about Healthcare field usage in the Extended Parameters-Healthcare and Healthcare Class sections. • Noted ReferenceNumber max length of 40 characters for Check/ACH transactions in Transaction Class section and CheckVerification, CheckSale, CheckCredit, CheckReturn, CheckVoid, CheckReversal, and CheckAppendImage methods. • Updated CreditCardCredit method to reflect OmniToken support (updating document to reflect current Express capabilities). • Updated Incremental Authorization section and CreditCardIncrementalAuthorization method to indicate support for Discover cards, as well as support for other industries such as Restaurants/Bars. • Removed Amount, AVS, and CVV test simulator tables and replaced with links to FIS Dev Engine. • Removed PASSUpdaterOption field from PaymentAccountCreate and PaymentAccountCreateWithTransID methods (not used for those methods)	3.1.2
2022-02-07	Jeff Gross	• Clarification note that for ReceiptEmail and ReceiptPhone functionality, a Beta Agreement is currently required. Also added legal notice regarding ReceiptEmail and ReceiptPhone under Transaction Class table. • Updated TransactionSetupID description and expiration information • Corrected ReversalType description in CheckReversal method • Added OmniToken use case and TokenOptions detail to use a prior OmniToken to generate a TokenRegistrationID • Updated description of QueryData in ScheduledTaskQuery method. Max number of records returned increased from 1,000 to 2,500. • Updated CreditCardSale method to include 'Other Optional Transaction Fields' section • Updated DebitCardPinlessSale and DebitCardPinlessReturn methods to include support with First Data-Nashville platform • Updated DebtRepaymentFlag description to reflect support for Vantiv, First Data-Nashville, and TSYS platforms	3.0.12
2021-05-12	Jeff Gross	• Updated FraudSightRiskStatus description • Updated links to Express specification on FIS Developer Engine • Updated terminology and email address for Developer Integrations team • Added notes to TokenRegistration description in Token Class • Added ReceiptEmail and ReceiptPhone input fields to Transaction Class for electronic transaction receipt delivery	3.0.11
2021-02-25	Jeff Gross	• Corrected internal document links to ExpressResponseCode, AVSResponseCode, and CVVResponseCode tables • Added DebtRepaymentFlag to Transaction Class as optional request parameter for use with Debt Repayment transactions • Updated AVSResponseCode descriptions • Updated CreditCardIncrementalAuthorization description and details • Removed compliance restriction note on Merchant Descriptor details, and Merchant Descriptor details can be included where supported going forward • Updated PAR response field to include support for FDC • Added TransactionID input option to EnhancedBINQuery method • Updated CheckQuery method description	3.0.10
2020-10-22	Jeff Gross	• Updated references to SOAP stating that it should no longer be used • Updated MotoECICode and Cryptogram input fields to indicate support for 3-D Secure 2.0 transactions (pass-through data only) • Clarified requirement of Healthcare Amount Type of 2/Healthcare when Healthcare/IIAS details are included for FSA/HSA cards	3.0.9

Date	Name	Change Description	Version
		• Clarified max amount of ConvenienceFeeAmount • Clarified ScheduledTaskRunLogID • Added pipe-delimited ACHNotes and ACHScores response fields to CheckQuery and CheckVerification methods. Document update only, as values are already returned for these methods. • Updated TransactionSetup Class with HPType (only used with Hosted Payments Premium integration) • Added Token Class option to TransactionSetup method (only used with Hosted Payments Premium integration) • Updated parameter name DuplicateCheckOverrideFlag to correct value of DuplicateOverrideFlag • Added FraudSightRiskStatus and FraudSightScore to Transaction Class as potential response parameters for FraudSight users • Added WebSessionID to Transaction Class as optional Ecommerce request parameter for use with FraudSight • Added ACH CheckAppendImage and Image Class in support of Check 21 Remote Deposit Capture (Future Use – Not Yet Available)	
2020-07-22	Jeff Gross	• Updated Account Updater details to reference support for Amex and Discover (acquired merchants only) • Added new PASSUpdaterStatus enumeration values in support of Amex and Discover • Added AccountType input parameter to TokenCreateWithTransID to support ACH token creation • Added clarification of ApprovedAmount response for CreditCardIncrementalAuthorization method	3.0.8
2020-05-18	Jeff Gross	• Updated CommercialCardResponseCode values for Visa "E" and "L" (Vantiv only) • Added DuplicateCheckExtended to Transaction Class • Corrected spelling of PINlessPOSConversionIndicator in Transaction Class • Added statement on AVS Response Codes table stating that AVS support outside the U.S. will be inconsistent. • Updated DebitCardPinlessSale and DebitCardPinlessReturn to reflect support for OmniToken • Removed Partial Reversal text under DebitCardReversal method • Added underscore character in sale example test card number to prevent blocking by security system • Updated Merchant Descriptor details to state that this functionality will be unavailable after 1/1/2021 • Updated PINless Debit methods to clarify that BillPaymentFlag should be set to 1/True • Updated PAR description in Transaction Class to indicate support for TSYS	3.0.7
2020-04-13	Jeff Gross	• Corrected DDAAccountType input parameter to indicate that it is required in CheckSale, CheckCredit, and CheckVerification methods • Corrected request format of PASSUpdaterDateTimeBegin and PASSUpdaterDateTimeEnd input fields in PaymentAccountQuery method • Added TerminalModel to Terminal Class • Noted that Merchant Descriptor functionality should no longer be implemented (per Network Compliance) • Corrected ScheduledTask and ScheduledTaskParameters Class hyperlinks • Updated HostBatchID max length from 25 to 10 characters and clarified that HostedBatchID values may be reused after exceeding an even smaller threshold • Added optional SECCode input field to CheckSale and CheckCredit methods. Added SECCode to Transaction Class. • Added pipe-delimited ACHNotes and ACHScores response fields to CheckSale method. Added ACHNotes and ACHScores to Transaction Class. • Added optional ACH CustomField[1-10] to Transaction Class • Added DeferredAuthIndicator to Transaction Class to support Store & Forward transactions • Added EncryptedFormat value of 11 in EncryptedFormat enumeration in support of Ingenico Generic TDES devices • Added ReferenceNumber to ScheduledTaskQuery method as optional query input parameter • Updated Scheduled Tasks text to indicate support of OmniToken during Scheduled Task creation	3.0.6
2020-01-20	Jeff Gross	• Added TokenRegistrationID description in Token Class	3.0.5

Date	Name	Change Description	Version
		• Updated TokenRegistrationID description in TokenCreate method • Added MonthlyLastDay (XML only) to the existing RunFrequency enumeration • Added EncryptedFormat value of 10 in EncryptedFormat enumeration in support of Ingenico Tetra devices • Corrected CheckNumber max length to be 8 characters • Updated BatchIndexCode to indicate that FirstPrevious is no longer valid • Clarified that XID and CAVV are future use only • Clarified max LineItemCount per transaction	
2019-08-09	Jeff Gross, Jerry Bevington, Jerry Hess	• Updated FIS Branding • Added ApprovalNumber for CreditCardForce in TransactionSetup	3.0.4
2019-07-03	Jeff Gross, Jerry Bevington	• Updated Adjustment transaction description	3.0.3
2019-05-02	Jeff Gross, Katharine Unsworth, Jerry Bevington	• Document redesign • Added PAR as possible response parameter • Added ACH support for TokenCreate • Added OmniToken support for ACH transaction methods • Added Ecommerce TypeOfGoods as optional input parameter • Added TSYS EMV direct certification support for HostApplicationID and HostDeveloperID input parameters • Added NetTranID and TranDT as possible Hosted Payments ReturnURL query string parameters	3.0.2
2019-02-15	Jeff Gross	• Corrected ExpressResponseMessage description for ExpressResponseCode 105 • Cleaned up Level III Line Item Detail text • Clarified CheckReversal (System) usage • Clarified required fields on Check/ACH transactions • Clarified use of TLS instead of SSL • Added Union Pay to possible CardLogo and PaymentBrand parameters • Clarified limited support of Auto Rental transactions in Hosted Payments (TransactionSetup). • Clarified PaymentAccountReferenceNumber definition • Cleaned up Account Updater details • Added Credential-on-File parameters to Transaction class • Added PaymentType and SubmissionType enumerations	2.9.5
2018-08-17	Jeff Gross	• Additional clarification on Terminal Type value	2.9.4
2018-08-07	Jeff Gross	• Clarified Hosted Payments LogoURL use • Updated card data retention information	2.9.3
2018-03-29	Chris Conrow Jeff Gross	• Added Soft Descriptor fields • Updated language for CreditCardVoid usage • Updated TransactionQuery description to call out one year non-sensitive transaction data retention policy	2.9.2
2018-03-23	Chris Conrow Jeff Gross	• Added CheckQuery transaction • Remove VaultID input field requirement for OmniToken.	2.9.1
2018-03-20	Wes Bowman	• Added NetworkLabel response parameter to transactions	2.9.0
2018-02-02	Kyle Ferrio Jeff Gross	• Added PINlessPOSConversionIndicator request and response parameters to transactions (Vantiv only) • Updated Hosted Payments response parameters to include BIN and Entry • Added TokenRegistrationID as optional input parameter on TokenCreate method	2.8.9
2018-01-15	Jeff Gross	• Updated TransactionID max length to 19 digis	2.8.8
2017-11-13	Kyle Ferrio	• Clarified the vehavior of AccountTokenCreate	2.8.7
2017-10-11	Jeff Gross	• Added BIN response parmeter to transactions • Updated MagneprintData max length • Corrected TerminalID to be listed as required in CreditCardSale	2.8.6
2017-09-14	Jeff Gross	• Added notes to SOAP references stating that we are no longer making enhancements to the SOAP interface • Added AddressEditAllowed Boolean parameter on TransactionSetup method	2.8.5
2017-08-23	Jeff Gross	• Added response parameters for Duplicate declines (XML only, terminal capture processors only)	2.8.4
2017-05-11	Xiaoming Du	• Added Surcharge changes	2.8.3
2017-04-19	Chris Conrow	• Added In-App (Transaction) changes	2.8.2
2017-04-18	Manish Anavkar	• Added OmniToken (Transaction) changes	2.8.1
2017-04-13	Dan Murphy	• Added OmniToken information for TokenCreate and TokenCreateWithTransID	2.8.0
2017-02-09	Dan Murphy Jeff Gross	• Added Dynamic Currency Conversion (DCC) information (XML only) • Added Hosted Payments CSS information (XML only)	2.7.9

Date	Name	Change Description	Version
2016-10-28	Dana Fowler Jeff Gross	• Added AutoRentalDailyRentalRate, AutoRentalWeeklyRentalRate, and AutoRentalInsuranceCharges	2.7.8

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

  Transactions Supported
    o   Business Rules and Process Flow for Transactions
    o   Security Methods and Procedures
    o   Developers and experienced technical people

- **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 transaction flow diagram followed by studying the Interface Specification. This is a quick way to help you get familiar with the interface elements.

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

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

worldpay | ONE   We highly recommend you periodically check for the latest version of this document at: https://developerengine.fisglobal.com/apis/express/express-xml

Product Overview

What is Worldpay's Express Platform?

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.

Payment Device ↔ Internet ↔ **worldpay** Express/Worldpay ↔ Card Brands

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, 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. Note that we are no longer making enhancements to the legacy SOAP interface, so it should not be used.

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

Express has the versatility to handle additional merchant types. If you do not see what you need, or have any questions, please contact: certification2@fisglobal.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/Payment Account Secure Storage (PASS)

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

Application Program Interface (API) Overview

Interfaces Supported

- 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.

  NOTE: Simple Object Access Protocol (SOAP)
  This legacy interface is no longer receiving development enhancements and therefore should not be used as an integration option.

  .

Transaction Flow

The below provides a high level view of the communication path when a merchant runs a POS payment transaction:
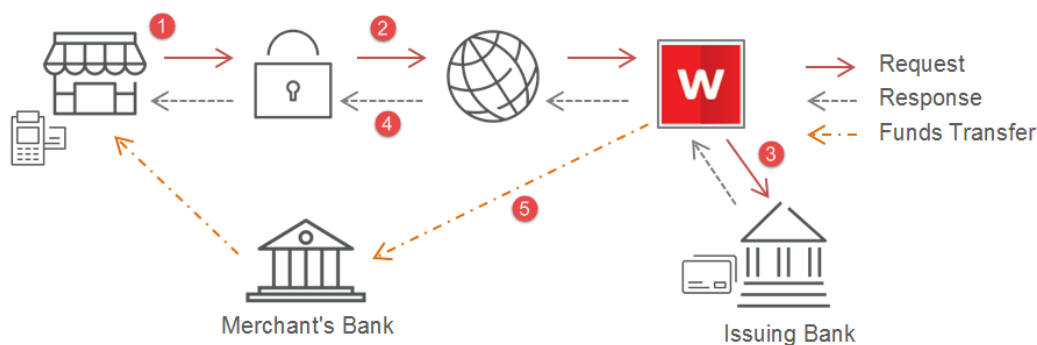
1. The merchant swipes the customer's card, which generates an authorization request based on the purchase price.
2. The request is sent via secured payment data along the internet to Express/Worldpay.
3. Express/Worldpay forwards the authorization request to the customer's card issuing bank. The issuing bank verifies the purchase against the customer's available credit amount, then generates either an "approval" or a "decline" authorization response to Express/Worldpay.
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, funds are transferred to the merchant's bank.

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.

Supported Transactions

Express supports Credit, Debit, Check, HealthCare (FSA/HAS), Electronic Benefits Transfer (EBT) and Gift. Credit, Debit, and Check/ACH transactions are described in the details below.

worldpay | ONE   Details and additional supplemental guides are available for EBT and Gift are on Worldpay O.N.E. athttps://developerengine.fisglobal.com/apis/express/express-xml

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.

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.

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.

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.

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. AI Responses – The AI Response (Approved with positive ID) usually results from a person using a card for the first time.

Void

A Void transaction can be used to back out a previous Sale transaction in the event a Reversal was unsuccessful. This applies to all transactions that were entered into the current batch, and only for the full dollar amount of the original transaction.

Adjustment

The Adjustment transaction is used to associate Level III Line Item Detail to a prior successful credit card transaction.

Incremental Authorization

Incremental authorization transactions are used to authorize an additional amount associated to a previous approved authorization, most commonly used in the Auto Rental and Lodging merchant industries. However, additional merchant types, such as Restaurant/Bars, now also support incremental authorizations. This method is used on the Vantiv, TSYS, FDC, and Global platforms. Visa, MasterCard, and Discover all support incremental authorizations.

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.

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 Developer Integrations team for further 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.

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.

Sale

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

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.

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.

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.

Query

A Query transaction is used to check on the status of a pending check transaction.

Append Image

An Append Image transaction is used to include check images (front and back) in support of Check 21 Remote Deposit Capture.

worldpay | ONE   Details and additional supplemental guides are available for Electronic Benefits Transfer (EBT), and Gift are on Worldpay O.N.E. athttps://developerengine.fisglobal.com/apis/express/express-xml

For details on Healthcare, Auto Rental, Lodging, and Level III Line Item Detail, please see the **Extended Parameters** section below.

Express Hosted Payments

How does it work?

Express offers two types of Hosted Payments:

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

For either solution, the software vendor's application is responsible for collecting all of the non-sensitive data needed to perform a payment transaction. 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.
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 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	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
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	LastFour	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", "Union Pay"
ApprovedAmount	ApprovedAmount	Approved amount (*depending on account type, ApprovedAmount may not be returned)
ServicesID	ServicesID	Numeric ServicesID
PaymentAccountID	PaymentAccountID	PaymentAccountID GUID
CommercialCardResponseCode	CommercialCardResponseCode	Commercial Card Response Code identifying type of commercial card if merchant profile supports this functionality
TipAmount	TipAmount	Tip Amount is returned if entered into driverless terminal device
BillingAddress1	BillingAddress1	BillingAddress1 is returned if submitted
BillingZipcode	BillingZipcode	BillingZipcode is returned if submitted
Bin	Bin	If AutoReturn=1/True, the first 6 digits of the card will be returned for financial transactions (not tokenization requests)
Entry	"Manual" or "Swipe" or "Chip"	Entry mode of the transaction processed
NetTranID	NetTranID	Express NetworkTransactionID (if returned by processor)
TranDT	TranDT	ExpressTransactionDate in "YYYY-MM-DD HH:MM:SS" format

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.

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. Also note that a TransactionSetupID will no longer be usable after five (5) unsuccessful attempts to process the transaction on the Hosted Payments page. Once that decline threshold is exceeded, the cardholder will see an error displayed on the Hosted Payments page, at which point the payment process must start again by obtaining a new TransactionSetupID.

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 select 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 (PASS)

TransForm Tokenization, also referred to as Payment Account Secure Storage (PASS), 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:

Method	Description
PaymentAccountCreate	The Create method inserts a new payment account record.
PaymentAccountUpdate	The Update method amends an existing payment account record.
PaymentAccountDelete	The Delete method removes an existing payment account record.
PaymentAccountQuery	The Query method retrieves an existing payment account record.

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.

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?

The Account Updater service provides merchants the ability to receive automatic updates to existing records stored with our Payment Account Secure Storage (PASS) and OmniToken products to prevent billing to expired or invalid cards. When an expired or invalid card is flagged, our system will query the payment brands for updated cardholder information. If the information is available, your customer's account on file (cardholder billing record) will be automatically updated and available for future billing.

How does Account Updater work?

Merchant locations must first enroll in our Account Updater service, which is designed to work through a software application integration to the Express platform. Software applications can be set to flag token records for update each time a token transaction is declined, and can also allow merchants to manually flag for update any token records 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 token records to determine if a card has been successfully updated. Updates are usually received within a couple business days (but could take up to approximately twelve business days), and we will not update a token record until an update response is received. If the issuer provides no update information after approximately twelve days, the Express platform will set the token status as "NoResponse" (enum value 10). If an update was successful, merchants can submit the updated token 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 token record on file.

Which cards can be flagged for update?

Previously, only Visa and MasterCard cards could be used with the Account Updater PaymentAccountAutoUpdate method, but beginning in July 2020, Express can also support the flagging of Discover and American Express cards for the service (with some restrictions). Please reference the Payment Account Auto Update Overview document from the Developer Integrations team.

More Information

For additional details regarding the Account Updater service, please contact your Developer Integrations implementation consultant or email certification2@fisglobal.com.

Security Methods and Procedures

Worldpay Integrated Payments, LLC, as a financial service provider, and the Express processing platform are 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.

Integrating Express 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.

Express requires all transactions to be sent using standard Secure Socket Layer (SSL) encryption. Additionally, every transaction must be authenticated. Authentication is achieved through 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.

Extended Parameters

This section provides descriptions of extended parameters for Healthcare, Auto Rental, Lodging, and Level III Line Item Detail.

Healthcare

Healthcare Extended Parameter Fields

Used when submitting a Healthcare transaction (Vantiv, TSYS, FDC, Global, or Paymentech platform only). Please note that Healthcare fields are only required for merchants/integrations that wish to accept FSA/HSA cards at merchant locations that also sell non-healthcare goods (such as with pharmacies and big-box retailers). Healthcare fields are not used to accept FSA/HSA cards for merchant locations that provide only healthcare goods and services (such as doctors offices and hospitals).

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

HealthcareFourthAmountType	ExtendedParameters.Healthcare	Healthcare amount type
HealthcareFourthCurrencyCode	ExtendedParameters.Healthcare	Currency Code. Use 840 for U.S. Leave blank if not used.
HealthcareFourthAmountSign	ExtendedParameters.Healthcare	Positive or negative amount
HealthcareFourthAmount	ExtendedParameters.Healthcare	Healthcare amount. Leave blank if not used.

Auto Rental

Auto Rental Extended Parameter Fields

Used when submitting an Auto Rental transaction (Vantiv, TSYS, FDC, or Global platform only).

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

Lodging

Lodging Extended Parameter Fields

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

Name	Class	Description
LodgingAgreementNumber	ExtendedParameters.Lodging	Hotel Folio number
LodgingCheckInDate	ExtendedParameters.Lodging	Check-in date. Format is YYYYMMDD. Advance check-in = date of reservation No-show = anticipated arrival date
LodgingCheckOutDate	ExtendedParameters.Lodging	Check-out date. Format is YYYYMMDD.
LodgingRoomAmount	ExtendedParameters.Lodging	Nightly rate for one room
LodgingRoomTax	ExtendedParameters.Lodging	Reserved for future use
LodgingNoShowIndicator	ExtendedParameters.Lodging	Indicates a room was reserved, but not actually rented. 0 = Not Applicable 1 = No Show

Name	Class	Description
LodgingDuration	ExtendedParameters.Lodging	Number of hotel stay days (e.g. 03). Pass 01 for no-show. From 01 to 99.
LodgingCustomerName	ExtendedParameters.Lodging	Lodging customer name
LodgingClientCode	ExtendedParameters.Lodging	Reserved for future use
LodgingExtraChargesDetail	ExtendedParameters.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	ExtendedParameters.Lodging	Reserved for future use
LodgingPrestigiousPropertyCode	ExtendedParameters.Lodging	Prestigious property code
LodgingSpecialProgramCode	ExtendedParameters.Lodging	Special program code
LodgingChargeType	ExtendedParameters.Lodging	Lodging charge type

EnhancedData (Level III Line Item Detail)

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

Name	Class	Max	Required	Description
MerchantVATRegistrationNumber	ExtendedParameters.EnhancedData	20	Optional	Government-assigned tax identification number of merchant. (tag must be present, but can be empty)
CustomerVATRegistrationNumber	ExtendedParameters.EnhancedData	13	Optional	Tax registration number supplied by cardholder. (tag must be present, but can be empty)
SummaryCommodityCode	ExtendedParameters.EnhancedData	4	Required	International description code of overall goods or services.
DiscountAmount	ExtendedParameters.EnhancedData	12	Required	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. DDDDDDDDD.CC
FreightAmount	ExtendedParameters.EnhancedData	12	Required	Freight or shipping portion of the total transaction amount. DDDDDDDDD.CC
DutyAmount	ExtendedParameters.EnhancedData	12	Required	Fee amount associated with the import of the goods. DDDDDDDDD.CC
DestinationZIPCode	ExtendedParameters.EnhancedData	10	Required	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	Postal/Zip Code of the address from where the purchased goods are being shipped.
DestinationCountryCode	ExtendedParameters.EnhancedData	3	Required	Country code where goods are being shipped. U.S. is 840.
UniqueVATInvoiceReferenceNumber	ExtendedParameters.EnhancedData	15	Optional	Invoice number associated with VAT invoice. Not applicable for US merchants. (tag must be present, but can be empty)
OrderDate	ExtendedParameters.EnhancedData	8	Required	Purchase order date. YYYYMMDD

Name	Class	Max	Required	Description
VATAmount	ExtendedParameters.EnhancedData	12	Required	Amount of any sales or value added taxes associated with the purchased item. DDDDDDDDD.CC
VATRate	ExtendedParameters.EnhancedData	4	Required	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	Number of line items included in transaction. Express supports max of 50 line items per transaction.
AlternateTaxAmount	ExtendedParameters.EnhancedData	12	Required	Second tax amount (used in countries where more than one tax can be applied). Not applicable for US merchants. DDDDDDDDD.CC
NationalTaxAmount	ExtendedParameters.EnhancedData	12	Required	Second tax amount (used in countries where more than one tax can be applied). Not applicable for US merchants. DDDDDDDDD.CC
ItemCommodityCode	ExtendedParameters.EnhancedData	12	Required	International description code of the individual good or service being supplied. Recommend using national or international list of standardized codes.
ItemDescription	ExtendedParameters.EnhancedData	35	Required	Item description.
ProductCode	ExtendedParameters.EnhancedData	12	Required	Merchant-defined description code of the item.
Quantity	ExtendedParameters.EnhancedData	12	Required	Quantity of the item (up to four decimal places). DDDDDDD.CCCC
UnitOfMeasure	ExtendedParameters.EnhancedData	12	Required	Code for units of measurement used in international trade.
UnitCost	ExtendedParameters.EnhancedData	10	Required	Unit cost of the item. DDDDDDD.CC
LineItemVATAmount	ExtendedParameters.EnhancedData	12	Required	Amount of any sales or value added taxes associated with the item.
LineItemVATRate	ExtendedParameters.EnhancedData	4	Required	Tax rate used to calculate LineItemVATAmount. Submitted as a percentage. Two-digit implied decimal. For example, if 10.42%, submit 1042
LineItemDiscountAmount	ExtendedParameters.EnhancedData	12	Required	Discount amount applied to the line item. This field should be 0.00 if the unit cost reflects a discount price or if no line item discount applies DDDDDDDDD.CC
LineItemTotalAmount	ExtendedParameters.EnhancedData	12	Required	Total full amount associated with the line item (including tax). DDDDDDDDD.CC
AlternateTaxIdentifier	ExtendedParameters.EnhancedData	15	Optional	Tax identification number of the merchant reporting the alternate tax amount. Not applicable for US merchants. (tag must be present, but can be empty)
VATType	ExtendedParameters.EnhancedData	4	Optional	Tax identification number of the merchant reporting the alternate tax amount. Not

Name	Class	Max	Required	Description
				applicable for US merchants. (tag must be present, but can be empty)
DiscountCode	ExtendedParameters.EnhancedData	2	Required	Indicates whether discount was applied to the purchase price of the item. 0 = Not supported 1 = Amount is discounted 2 = Amount is not discounted
NetGrossCode	ExtendedParameters.EnhancedData	2	Required	Indicates whether extended item amount included the tax. 0 = Not supported 1 = Item Amount includes tax amount 2 = Item Amount does not include tax amount
ExtendedItemAmount	ExtendedParameters.EnhancedData	12	Required	Amount of the item (price multiplied by quantity). DDDDDDDDD.CC
DebitCreditCode	ExtendedParameters.EnhancedData	2	Required	Indicates whether 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.EnhancedData	5	Required	Discount rate for the line item. Submitted as a percentage. Two-digit implied decimal. For example, if 10.42%, submit 1042

Level III XML Format

Note: 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 />
    <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>
```

Express 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.

Since all per-transaction duplicate checking enable / disable functionality can be accomplished with the single DuplicateCheckDisableFlag parameter, it is recommended that integrators move forward with implementing only the DuplicateCheckDisableFlag parameter.

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. You may pass a TokenID, TokenProvider, CardLogo, and Expiration Date is OmniToken Tokenization is used. If you choose to pass in the original card or payment information, Express will automatically create a PaymentAccount 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.

Dynamic Currency Conversion (XML only)

For Express/Vantiv gateway clients only, Express supports the ability for Dynamic Currency Conversion (DCC) that informs the merchant to prompt the consumer to ask if they want to be paid in their native currency. This is a two-pass transaction process. The first pass transaction will respond with ExpressResponseCode = 7 indicating the merchant must prompt the consumer for a currency.

Processing flow is defined below.

Sale Transactions

1. First Pass: The merchant submits a Sale to Express in normal fashion. Express may respond with an ExpressResponseCode of 7 and ExpressResponseMessage of "DCCRequested". This means that the transaction did not authorize and the merchant must prompt the user to see if they want to pay in their native currency.
2. Merchant Prompts Consumer: The first pass response contains 3 fields in the <Transaction> section as listed below that allows the Merchant to prompt the consumer with the foreign currency, foreign transaction amount, and the foreign exchange rate.

   <ForeignTransactionAmount>1.80</ForeignTransactionAmount>

   <ForeignCurrencyCode>036</ForeignCurrencyCode>

   <ConversionRate>61370638</ConversionRate>

3. Second Pass: The merchant submits the Sale in a second pass transaction that indicates if the consumer opted in or opted out. The fields passed in the request for the second pass transaction are in the <Transaction> section as follows below. DCCRequested is 1 to opt in and 0 to opt out. Even if the consumer opts out, these 4 fields must be sent. If the transaction amount is adjusted, the merchant is responsible for applying the conversion rate to calculate a new ForeignTransactionAmount.

   <DCCRequested>1</DCCRequested>

   <ConversionRate>61370638</ConversionRate>

   <ForeignCurrencyCode>036</ForeignCurrencyCode>

   <ForeignTransactionAmount>1.80</ForeignTransactionAmount>

4. The 2nd pass transaction should approve as normal.

Custom CSS (XML only)

Hosted Payments has the ability to allow custom styling of the payment page. This is accomplished by passing custom CSS in the CustomCss field in the TransactionSetup method. This allows the integrator to match the style of the payment page with the merchant's style. By doing so, it creates a more seamless user experience which enhances consumer trust when the payment is processed.

The CSS that can be passed into the CustomCss field is limited for security reasons. See the Custom CSS appendix for details on what is allowed along with a simple example.

Level III Processing (XML only)

Express supports the ability to process Visa and MasterCard Level III Enhanced and Line Item Detail. Processing flow is defined below.

Sale Transactions (charge immediately)

1. 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)

1. 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.

Additional notes on Level III

- 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.
- 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.

Express Methods

Batch Methods

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:

- **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.

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	Merchant ID
TerminalID	Terminal	Required	Unique terminal identifier
BatchCloseType	Batch	Required	Type of batch close
HostBatchID	Batch	Required	Unique host batch identifier
HostBatchCount	Batch	Conditional	Batch count. Used when BatchCloseType = Regular
HostCreditSaleCount	Batch	Conditional	Total numeric count of credit sale items. Used when BatchCloseType = Regular
HostCreditSaleAmount	Batch	Conditional	Total dollar amount of credit sales with decimal place. Example: 21.52. Used when BatchCloseType = Regular
HostCreditReturncount	Batch	Conditional	Total numeric count of credit return items. Used when BatchCloseType = Regular
HostCreditReturnAmount	Batch	Conditional	Total dollar amount of credit returns with decimal place. Example: 21.52. Used when BatchCloseType = Regular
HostDebitSaleCount	Batch	Conditional	Total numeric count of debit sale items. Used when BatchCloseType = Regular
HostDebitSaleAmount	Batch	Conditional	Total dollar amount of debit sales with decimal place. Example: 21.52. Used when BatchCloseType = Regular
HostDebitReturnCount	Batch	Conditional	Total numeric count of debit return items. Used when BatchCloseType = Regular
HostDebitReturnAmount	Batch	Conditional	Total dollar amount of debit returns with decimal place. Example: 21.52. Used when BatchCloseType = Regular

Output Fields

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	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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostBatchCount	Response.Batch	Returned	Total numeric count of all items in the batch
HostBatchAmount	Response.Batch	Returned	Total dollar amount of all items in the batch
HostCreditSaleCount	Response.Batch	Returned	Total numeric count of credit sale items
HostCreditSaleAmount	Response.Batch	Returned	Total dollar amount of credit sales
HostCreditReturnCount	Response.Batch	Returned	Total numeric count of credit return items
HostCreditReturnAmount	Response.Batch	Returned	Total dollar amount of credit returns
HostDebitSaleCount	Response.Batch	Returned	Total numeric count of debit sale items
HostDebitSaleAmount	Response.Batch	Returned	Total dollar amount of debit sales
HostDebitReturnCount	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.

BatchTotalsQuery

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	Merchant ID
TerminalID	Terminal	Required	Unique terminal identifier
BatchQueryType	Batch	Required	Type of batch query
HostBatchID	Batch	Optional	If null, current batch is used
BatchIndexCode	Batch	Optional	Identifies where current or previous batches are used

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostBatchCount	Response.Batch	Returned	Total numeric count of all items in the batch
HostBatchAmount	Response.Batch	Returned	Total dollar amount of all items in the batch
HostCreditSaleCount	Response.Batch	Returned	Total numeric count of credit sale items
HostCreditSaleAmount	Response.Batch	Returned	Total dollar amount of credit sales
HostCreditReturnCount	Response.Batch	Returned	Total numeric count of credit return items
HostCreditReturnAmount	Response.Batch	Returned	Total dollar amount of credit returns
HostDebitSaleCount	Response.Batch	Returned	Total numeric count of debit sale items
HostDebitSaleAmount	Response.Batch	Returned	Total dollar amount of debit sales
HostDebitReturnCount	Response.Batch	Returned	Total numeric count of debit return items
HostDebitReturnAmount	Response.Batch	Returned	Total dollar amount of debit returns
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	State/status of transaction

Notes on BatchTotalsQuery

- BatchTotalsQuery will only return complete totals for the current open batch.

Credit Methods

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.

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
PartialApprovedFlag	Transaction	Required	Specifies if POS supports partial approvals
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
PaymentType	Transaction	Optional	Payment type used for credential on file transactions
SubmissionType	Transaction	Optional	Submission type used for credential on file transactions
NetworkTransactionID	Transaction	Optional	Transaction identifier used for credential on file transactions
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional [1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional [1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
Track1Data	Card	Conditional [1]	Track1 Data

Name	Class	Required	Description
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
CVV	Card	Optional	CVV2 / CVC2 / CID. May be submitted for Keyed transactions.
[Other Optional Card Fields]	Card	Optional	See Card Class.
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
[Address Fields]	Address	Optional	See Address Class. For keyed transactions, BillingAddress1 and BillingZipcode should be included.
[Scheduled Task Fields]	ExtParm.ScheduledTask	Optional	See ScheduledTask Class
[Healthcare Fields]	ExtParm.Healthcare	Optional	See Healthcare Class. May be required for FSA / HSA cards.
[Auto Rental Fields]	ExtParm.AutoRental	Optional	See AutoRental Class. For Auto Rental transactions, all fields are required.
[Lodging Fields]	ExtParm.Lodging	Optional	See Lodging Class. For Lodging transactions all fields are required.
[Level III Fields]	ExtParm.EnhancedData	Optional	See Level III EnhancedData section. For Level III transactions, all fields are required.
[Token Fields]	ExtParm.Token	Optional	See Token Class

Conditional[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.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
AVSResponseCode	Response.Card	Conditional	Address Verification service response
CVVResponseCode	Response.Card	Conditional	CVV2 / CVC2 / CID response
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
CommercialCardResponseCode	Response.Transaction	Conditional	Commercial card response code
BalanceAmount	Response.Transaction	Conditional	Balance amount
BalanceCurrencyCode	Response.Transaction	Conditional	Balance amount currency code
SurchargeAmount	Response.Transaction	Conditional	Returned if supplied
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)

Name	Class	Returned	Description
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)
NetworkTransactionID	Response.Transaction	Conditional	Transaction identifier used for credential on file transactions
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class
PaymentAccountID	Response.PaymentAccount	Conditional	Returned if ScheduledTask class was supplied
PaymentAccountReferenceNumber	Response.PaymentAccount	Conditional	Returned if ScheduledTask class was supplied
ScheduledTaskID	Response.ScheduledTask	Conditional	Returned if ScheduledTask class was supplied
ScheduledTaskName	Response.ScheduledTask	Conditional	Returned if ScheduledTask class was supplied
ScheduledTaskGroupID	Response.ScheduledTask	Conditional	Returned if ScheduledTask class was supplied
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied
TokenID	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TokenProvider	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TAProviderID	Response.Token	Conditional	Returned if TransArmor tokenization is used

CreditCardAuthorization

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

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
PartialApprovedFlag	Transaction	Required	Specifies if POS supports partial approvals
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
PaymentType	Transaction	Optional	Payment type used for credential on file transactions
SubmissionType	Transaction	Optional	Submission type used for credential on file transactions
NetworkTransactionID	Transaction	Optional	Transaction identifier used for credential on file transactions
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional [1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional [1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.

Name	Class	Required	Description
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
Track1Data	Card	Conditional [1]	Track1 Data
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
CVV	Card	Optional	CVV2 / CVC2 / CID. May be submitted for Keyed transactions.
[Other Optional Card Fields]	Card	Optional	See Card Class.
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
[Address Fields]	Address	Optional	See Address Class. For keyed transactions, BillingAddress1 and BillingZipcode should be included.
[Healthcare Fields]	ExtParm.Healthcare	Optional	See Healthcare Class. May be required for FSA / HSA cards.
AutoRentalDuration	ExtParm.AutoRental	Optional	See AutoRental Class. Required for Auto Rental transactions.
LodgingDuration	ExtParm.Lodging	Optional	See Lodging Class. Required for Lodging transactions.
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Optional	See Lodging Class. Required for Lodging transactions.
[Level III Fields]	ExtParm.EnhancedData	Optional	See Level III EnhancedData section. For Level III transactions, all fields are required.
[Token Fields]	ExtParm.Token	Optional	See Token Class

Conditional[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.

Output Fields

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
AVSResponseCode	Response.Card	Conditional	Address Verification service response
CVVResponseCode	Response.Card	Conditional	CVV2 / CVC2 / CID response
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
CommercialCardResponseCode	Response.Transaction	Conditional	Commercial card response code
BalanceAmount	Response.Transaction	Conditional	Balance amount
BalanceCurrencyCode	Response.Transaction	Conditional	Balance amount currency code
SurchargeAmount	Response.Transaction	Conditional	Returned if supplied
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)

Name	Class	Returned	Description
NetworkTransactionID	Response.Transaction	Conditional	Transaction identifier used for credential on file transactions
PaymentAccountID	Response.PaymentAccount	Conditional	Returned if ScheduledTask class was supplied
PaymentAccountReferenceNumber	Response.PaymentAccount	Conditional	Returned if ScheduledTask class was supplied
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied
TokenID	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TokenProvider	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TAProviderID	Response.Token	Conditional	Returned if TransArmor tokenization is used

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.

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
PaymentType	Transaction	Optional	Payment type used for credential on file transactions
SubmissionType	Transaction	Optional	Submission type used for credential on file transactions
NetworkTransactionID	Transaction	Optional	Transaction identifier used for credential on file transactions
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
[Auto Rental Fields]	ExtParm.AutoRental	Optional	See AutoRental Class. For Auto Rental transactions, all fields are required.

Name	Class	Required	Description
[Lodging Fields]	ExtParm.Lodging	Optional	See Lodging Class. For Lodging transactions all fields are required.
[Level III Fields]	ExtParm.EnhancedData	Optional	See Level III EnhancedData section. For Level III transactions, all fields are required.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)

CreditCardCredit

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

- **Example:** A customer returns merchandise and wants a refund.

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)

Name	Class	Required	Description
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional [1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional [1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
Track1Data	Card	Conditional [1]	Track1 Data
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
[Other Optional Card Fields]	Card	Optional	See Card Class.
[Token Fields]	ExtParm.Token	Optional	See Token Class.

Conditional[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.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Conditional	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)

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 45 days old.
- **Example:** A customer returns merchandise and wants a refund.

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Conditional	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)

CreditCardAdjustment

The Adjustment transaction is used to associate Level III Line Item Detail to a prior successful credit card transaction.

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
[Level III Fields]	ExtParm.EnhancedData	Required	See Level III EnhancedData section. For Level III transactions, all fields are required.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
TransactionID	Response.Transaction	Returned	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction

CreditCardVoid

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

- A Void should be used to remove an offline transaction from the batch, such as CreditCardReturn, CreditCardCredit, and CreditCardForce.

- It can also be used to remove a CreditCardAuthorizationCompletion from a batch when the intention is to submit a new Completion afterward.

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
TransactionID	Response.Transaction	Returned	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status 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.

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier
ApplicationName	Application	Required	Name of application
ApplicationVersion	Application	Required	Version of application

Name	Class	Required	Description
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount (use 0.00)
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
PaymentType	Transaction	Optional	Payment type used for credential on file transactions
SubmissionType	Transaction	Optional	Submission type used for credential on file transactions
NetworkTransactionID	Transaction	Optional	Transaction identifier used for credential on file transactions
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional [1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional [1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
Track1Data	Card	Conditional [1]	Track1 Data
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
CVV	Card	Optional	CVV2 / CVC2 / CID. May be submitted for Keyed transactions.
[Other Optional Card Fields]	Card	Optional	See Card Class.
BillingZipcode	Address	Required	The zip code used for billing purposes
BillingAddress1	Address	Required	The street address used for billing purposes
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
[Token Fields]	ExtParm.Token	Optional	See Token Class

Conditional [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.

Output Fields

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
AVSResponseCode	Response.Card	Conditional	Address Verification service response

Name	Class	Returned	Description
CVVResponseCode	Response.Card	Conditional	CVV2 / CVC2 / CID response. May be returned depending on processor.
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
CommercialCardResponseCode	Response.Transaction	Conditional	Commercial card response code
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)
NetworkTransactionID	Response.Transaction	Conditional	Transaction identifier used for credential on file transactions
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied
TokenID	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TokenProvider	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TAProviderID	Response.Token	Conditional	Returned if TransArmor tokenization is used

CreditCardReversal

There are three types of Reversal transactions:

- 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/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.

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	Merchant ID
ReversalType	Transaction	Required	Type of reversal
TransactionID	Transaction	Conditional	If ReversalType = Full or Partial, then include original TransactionID. If ReversalType = System, then do not include original TransactionID and use Card Class instead
TransactionAmount	Transaction	Required	If ReversalType = Full or System, then use original transaction amount. If ReversalType = Partial, then use new (replacement) amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number

Name	Class	Required	Description
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional [1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional [1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
Track1Data	Card	Conditional [1]	Track1 Data
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
[Other Optional Card Fields]	Card	Optional	See Card Class.
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
[Token Fields]	ExtParm.Token	Optional	See Token Class

Conditional [1] Only one of the following field groups needs to be included if ReversalType = System: 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	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
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
TokenID	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TokenProvider	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TAProviderID	Response.Token	Conditional	Returned if TransArmor tokenization is used

Additional notes on Reversals

Processing System Reversals:

1. **CreditCardSale** If you did not receive a response from a CreditCardSale transaction, call the CreditCardReversal method with a ReversalType of "System".

2. **CreditCardCredit, CreditCardReturn, CreditCardAuthorizationCompletion** If you did not receive a response from a call to one of these methods, you should resend the transaction.
   - 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:

- Timeout – If the merchant is unable to perform an online transaction.
- Referral – If the merchant receives an online referral message in response to an online transaction (i.e. Call Issuer, Call AMEX, etc).
- AI Responses – The AI response (Approved with positive ID) usually results from a person using a card for the first time.

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ApprovalNumber	Transaction	Required	Original approval number obtained during voice authoirization. Use capital letters and numbers only.
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional [1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional [1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data

Name	Class	Required	Description
Track1Data	Card	Conditional [1]	Track1 Data
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
[Other Optional Card Fields]	Card	Optional	See Card Class.
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
[Address Fields]	Address	Optional	See Address Class. For keyed transactions, BillingAddress1 and BillingZipcode should be included.
[Auto Rental Fields]	ExtParm.AutoRental	Optional	See AutoRental Class. For Auto Rental transactions, all fields are required.
[Lodging Fields]	ExtParm.Lodging	Optional	See Lodging Class. For Lodging transactions all fields are required.
[Level III Fields]	ExtParm.EnhancedData	Optional	See Level III EnhancedData section. For Level III transactions, all fields are required.
[Token Fields]	ExtParm.Token	Optional	See Token Class

Conditional [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.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied
TokenID	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TokenProvider	Response.Token	Conditional	Returned if OmniToken or TransArmor tokenization is used
TAProviderID	Response.Token	Conditional	Returned if TransArmor tokenization is used

CreditCardIncrementalAuthorization

CreditCardIncrementalAuthorization transactions are used to authorize an additional amount associated to a previous approved authorization, most commonly used in the Auto Rental and Lodging merchant industries. However, additional merchant types, such as Restaurant/Bars, now also support incremental authorizations. This method is used on the Vantiv, TSYS, FDC, and Global platforms. Visa, MasterCard, and Discover all support incremental authorizations.

- This method is used on the Vantiv, TSYS, FDC or Global platforms.
- Visa, MasterCard, and Discover all support incremental authorizations. Contact Developer Integrations for more information.

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
AutoRentalDuration	ExtParm.AutoRental	Optional	See AutoRental Class. Required for Auto Rental transactions.
LodgingDuration	ExtParm.Lodging	Optional	See Lodging Class. Required for Lodging transactions.
LodgingPrestigiousPropertyCode	ExtParm.Lodging	Optional	See Lodging Class. Required for Lodging transactions.

Output Fields

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
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction

Name	Class	Returned	Description
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount. Amount may vary by processor. Integrator application should track total approved amount separately.

Debit Methods

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.

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
PartialApprovedFlag	Transaction	Required	Specifies if POS supports partial approvals
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
CashBackAmount	Transaction	Optional	Cash back amount
PaymentType	Transaction	Optional	Payment type used for credential on file transactions
SubmissionType	Transaction	Optional	Submission type used for credential on file transactions
NetworkTransactionID	Transaction	Optional	Transaction identifier used for credential on file transactions
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
KeySerialNumber	Card	Required	Key Serial Number
PINBlock	Card	Required	Encrypted PIN Block
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.

Name	Class	Required	Description
Track2Data	Card	Conditional [1]	Track2 Data
[Other Optional Card Fields]	Card	Optional	See Card Class.
[Address Fields]	Address	Optional	See Address Class

Conditional [1] Only one of the following field groups needs to be included: Track2Data **or** MagneprintData **or** EncryptedTrack2Data.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
BalanceAmount	Response.Transaction	Conditional	Balance amount
BalanceCurrencyCode	Response.Transaction	Conditional	Balance amount currency code
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied

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 Developer Integrations team for further details.

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	Merchant ID

Name	Class	Required	Description
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
KeySerialNumber	Card	Required	Key Serial Number
PINBlock	Card	Required	Encrypted PIN Block
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
[Other Optional Card Fields]	Card	Optional	See Card Class.
[Address Fields]	Address	Optional	See Address Class

Conditional [1] Only one of the following field groups needs to be included: Track2Data **or** MagneprintData **or** EncryptedTrack2Data.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
BalanceAmount	Response.Transaction	Conditional	Balance amount
BalanceCurrencyCode	Response.Transaction	Conditional	Balance amount currency code
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)

Name	Class	Returned	Description
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied

DebitCardReversal

There are two types of Debit Reversal transactions:

1. 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.

2. 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.

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	Merchant ID
ReversalType	Transaction	Required	Type of reversal
TransactionID	Transaction	Conditional	If ReversalType = Full, then include original TransactionID. If ReversalType = System, then do not include original TransactionID and use Card Class instead
TransactionAmount	Transaction	Required	If ReversalType = Full or System, then use original transaction amount.
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year

Name	Class	Required	Description
[Other Optional Card Fields]	Card	Optional	See Card Class.

Conditional [1] Only one of the following field groups needs to be included if ReversalType = System: CardNumber / ExpirationMonth / ExpirationYear **or** Track2Data **or** MagneprintData **or** EncryptedTrack2Data

Output Fields

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
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)

Additional Notes on DebitCardReversals

Processing System Reversals

1. **DebitCardSale:** If you did not receive a response from a DebitCardSale transaction, call the DebitCardReversal method with a ReversalType of "System".

2. **DebitCardReturn:** If you did not receive a response from a DebitCardReturn, you should resend the transaction.
   - 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.

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 and First Data-Nashville platforms at this time. Please contact your Worldpay sales rep for additional information.

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	Merchant ID

Name	Class	Required	Description
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
BillPayerAccountNumber	Transaction	Required	Bill payer account number
BillPaymentFlag	Transaction	Required	Bill payment flag (should be set to 1/True)
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
EncryptedCardData	Card	Conditional [1]	Encrypted Track2 Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
[Other Optional Card Fields]	Card	Optional	See Card Class.
[Address Fields]	Address	Optional	See Address Class
[Token Fields]	ExtParm.Token	Optional	See Token Class

Conditional [1] Only one of the following field groups needs to be included: EncryptedCardData **or** CardNumber / ExpirationMonth / ExpirationYear.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
BalanceAmount	Response.Transaction	Conditional	Balance amount

Name	Class	Returned	Description
BalanceCurrencyCode	Response.Transaction	Conditional	Balance amount currency code
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied

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 for 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 and First Data-Nashville platforms at this time. Please contact your Worldpay sales rep for additional information.

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
BillPayerAccountNumber	Transaction	Required	Bill payer account number
BillPaymentFlag	Transaction	Required	Bill payment flag (should be set to 1/True)
[Other Optional Transaction Fields]	Transaction	Optional	See Transaction Class
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
EncryptedCardData	Card	Conditional [1]	Encrypted Track2 Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
[Other Optional Card Fields]	Card	Optional	See Card Class.
[Address Fields]	Address	Optional	See Address Class

Name	Class	Required	Description
[Token Fields]	ExtParm.Token	Optional	See Token Class

Conditional [1] Only one of the following field groups needs to be included: EncryptedCardData **or** CardNumber / ExpirationMonth / ExpirationYear.

Output Fields

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
HostBatchID	Response.Batch	Returned	Unique host batch identifier
HostItemID	Response.Batch	Returned	Host item identifier
HostBatchAmount	Response.Batch	Returned	Total amount of items in batch
CardLogo	Response.Card	Returned	Payment brand
BIN	Response.Card	Returned	Card BIN (first 6 digits)
TransactionID	Response.Transaction	Returned	Transaction identifier
ApprovalNumber	Response.Transaction	Returned	Issuer-assigned approval number
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
AcquirerData	Response.Transaction	Returned	Acquire reference data. No action needed.
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ApprovedAmount	Response.Transaction	Conditional	Approved amount
BalanceAmount	Response.Transaction	Conditional	Balance amount
BalanceCurrencyCode	Response.Transaction	Conditional	Balance amount currency code
DuplicateTransactionID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
DuplicateHostItemID	Response.Transaction	Conditional	May be returned with duplicate decline (XML only)
NetworkLabel	Response.Transaction	Conditional	Network label (depends on processor)
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class
BillingAddress1	Response.Address	Conditional	Returned if supplied
BillingZipcode	Response.Address	Conditional	Returned if supplied

Check Methods

CheckVerification

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

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount

Name	Class	Required	Description
ReferenceNumber	Transaction	Required	Reference number. <mark>For Check/ACH transactions, ReferenceNumber max length is 40 characters.</mark>
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number
AccountNumber	DemandDepositAccount	Conditional [1]	Account Number of consumer bank account
RoutingNumber	DemandDepositAccount	Conditional [1]	Routing Number of consumer bank account
CheckNumber	DemandDepositAccount	Optional	Check Number
CheckType	DemandDepositAccount	Optional	Check Type
DDAAccountType	DemandDepositAccount	Required	DDA Account Type
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
BillingName	Address	Required	Consumer Name in FirstName LastName format
[Address Fields]	Address	Optional	See Address Class
[Identification Fields]	Identification	Optional	See Identification Class
[Token Fields]	ExtParm.Token	Optional	See Token Class. If used, TokenID and TokenProviderID are required.

Conditional [1] Only one of the following field groups needs to be included: AccountNumber / RoutingNumber **or** PaymentAccountID. Not included if token is submitted.

Output Fields

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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ACHNotes	Response.Transaction	Conditional	Pipe-delimited ACH notes. May be returned depending on Check Commerce configuration.
ACHScores	Response.Transaction	Conditional	Pipe-delimited ACH scores. May be returned depending on Check Commerce configuration.
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class

CheckSale

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

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	Merchant ID

Name	Class	Required	Description
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number. For Check/ACH transactions, ReferenceNumber max length is 40 characters.
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
SECCode	Transaction	Optional	Standard Entry Class code, such as "ARC", "BOC", "RCK", "CCD", "PPD", "POP", "TEL", or "WEB".
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number
AccountNumber	DemandDepositAccount	Conditional [1]	Account Number of consumer bank account
RoutingNumber	DemandDepositAccount	Conditional [1]	Routing Number of consumer bank account
CheckNumber	DemandDepositAccount	Optional	Check Number
CheckType	DemandDepositAccount	Optional	Check Type
DDAAccountType	DemandDepositAccount	Required	DDA Account Type
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
BillingName	Address	Required	Consumer Name in FirstName LastName format
[Address Fields]	Address	Optional	See Address Class
[Identification Fields]	Identification	Optional	See Identification Class
[Scheduled Task Fields]	ExtParm.ScheduledTask	Optional	See ScheduledTask Class
[Token Fields]	ExtParm.Token	Optional	See Token Class. If used, TokenID and TokenProviderID are required.

Conditional [1] Only one of the following field groups needs to be included: AccountNumber / RoutingNumber **or** PaymentAccountID. Not included if TokenID is submitted.

Output Fields

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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
ACHNotes	Response.Transaction	Conditional	Pipe-delimited ACH notes. May be returned depending on Check Commerce configuration.
ACHScores	Response.Transaction	Conditional	Pipe-delimited ACH scores. May be returned depending on Check Commerce configuration.
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class
ScheduledTaskID	Response.ScheduledTask	Conditional	Returned if ScheduledTask class was supplied
ScheduledTaskName	Response.ScheduledTask	Conditional	Returned if ScheduledTask class was supplied
ScheduledTaskGroupID	Response.ScheduledTask	Conditional	Returned if ScheduledTask class was supplied

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.

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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number. For Check/ACH transactions, ReferenceNumber max length is 40 characters.
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
SECCode	Transaction	Optional	Standard Entry Class code, such as "ARC", "BOC", "RCK", "CCD", "PPD", "POP", "TEL", or "WEB".
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number
AccountNumber	DemandDepositAccount	Conditional [1]	Account Number of consumer bank account
RoutingNumber	DemandDepositAccount	Conditional [1]	Routing Number of consumer bank account
CheckNumber	DemandDepositAccount	Optional	Check Number
CheckType	DemandDepositAccount	Optional	Check Type
DDAAccountType	DemandDepositAccount	Required	DDA Account Type
PaymentAccountID	ExtParm.PaymentAccount	Conditional [1]	Payment Account Identifier
BillingName	Address	Required	Consumer Name in FirstName LastName format
[Address Fields]	Address	Optional	See Address Class
[Identification Fields]	Identification	Optional	See Identification Class
[Token Fields]	ExtParm.Token	Optional	See Token Class. If used, TokenID and TokenProviderID are required.

Conditional [1] Only one of the following field groups needs to be included: AccountNumber / RoutingNumber **or** PaymentAccountID. Not included if token is submitted.

Output Fields

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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class

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.

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number. For Check/ACH transactions, ReferenceNumber max length is 40 characters.
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number
[Address Fields]	Address	Optional	See Address Class

Output Fields

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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class

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	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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
ReferenceNumber	Transaction	Required	Reference number. For Check/ACH transactions, ReferenceNumber max length is 40 characters.
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier

Name	Class	Required	Description
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number

Output Fields

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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class

CheckReversal

There is one type of Reversal available for Check transactions: **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.
- **NOTE:** CheckReversal require the ReferenceNumber from the original transaction to match/identify the transaction to reverse. AccountNumber/RoutingNumber are not used in the CheckReversal matching logic.

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	Merchant ID
ReversalType	Transaction	Required	Type of Reversal. Only System/0 is supported for CheckReversal.
ReferenceNumber	Transaction	Required	Reference number. For Check/ACH transactions, ReferenceNumber max length is 40 characters.
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number

Output Fields

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

Name	Class	Returned	Description
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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class

CheckQuery

CheckQuery is used to confirm the status of a prior CheckSale, CheckCredit, or CheckReturn transaction using a real-time request to the ACH/Check provider.

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	Merchant ID
TransactionID	Transaction	Conditional [1]	TransactionID returned in original transaction
ReferenceNumber	Transaction	Conditional [1]	Reference number
MarketCode	Transaction	Required	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
DDAAccountType	DemandDepositAccount	Conditional	Include DDAAccountType from original ACH transaction. 0 = Checking Account 1 = Savings Account
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number

Conditional [1] A transaction can be queried by either TransactionID or ReferenceNumber.

Output Fields

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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of status/state of transaction
TransactionStatusCode	Response.Transaction	Returned	State/status of transaction
NOCDate	Response.Transaction	Conditional [1]	Date of NOC
FundedDate	Response.Transaction	Conditional [1]	Date of transaction funding
DateProcessed	Response.Transaction	Returned	Date original transaction was run
ReturnDate	Response.Transaction	Conditional [1]	Date a return was processed

Name	Class	Returned	Description
ACHNotes	Response.Transaction	Conditional	Pipe-delimited ACH notes. May be returned depending on Check Commerce configuration.
ACHScores	Response.Transaction	Conditional	Pipe-delimited ACH scores. May be returned depending on Check Commerce configuration.
RCode	Response.Transaction	Conditional	ACH Return Reason Code. May be returned depending on Check Commerce configuration.
BOE	Response.Transaction	Conditional	Back-Office Exception Code. May be returned depending on Check Commerce configuration.
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class

Conditional[1] These fields are not returned in all cases and will be returned based on the status of the original transaction.

CheckAppendImage (Future Use)

A CheckAppendImage request is used to include a check image (front and/or back) in support of Check 21 Remote Deposit Capture for a prior CheckSale transaction.

- To include a check image, a prior CheckSale must exist. Use the original CheckSale TransactionID with this method to attach a front and back image of the check.

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
ReferenceNumber	Transaction	Optional	Reference number. For Check/ACH transactions, ReferenceNumber max length is 40 characters.
MarketCode	Transaction	Optional	Type of industry
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number
FileName	Image	Required	File name
ImageData	Image	Required	Byte Array image of the front or back of the check. This is the actual data of the check. You can submit either Tiff or Jpeg formatted data. The FileName's extension should match, or your file may not be submitted correctly. Jpeg format is the default.
ImageSide	Image	Required	Identifies the side of the check image. Allowed values are "Front" or "Back".

Output Fields

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	Transaction identifier
ReferenceNumber	Response.Transaction	Conditional	Returned if supplied

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	State/status of transaction
[Other Transaction Response Fields]	Response.Transaction	Conditional	See Transaction Class

Reporting Methods

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
- A maximum of 1000 records returned.
- Non-Sensitive transaction data is available up to 365 days.

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	Merchant ID
TransactionDateTimeBegin	Parameters	Conditional	Begin date/time of transaction range. Required if TransactionID or TransactionSetupID is not included. Format "YYYY-MM-DD HH:MM:SS"
TransactionDateTimeEnd	Parameters	Conditional	End date/time of transaction range. Required if TransactionID or TransactionSetupID is not included. Format "YYYY-MM-DD HH:MM:SS"
TransactionID	Parameters	Optional	TransactionID returned in original transaction
TerminalID	Parameters	Optional	Terminal identifier
ApplicationID	Parameters	Optional	Unique application identifier
ApprovalNumber	Parameters	Optional	Approval number
ApprovedAmount	Parameters	Optional	Approved amount
ExpressTransactionDate	Parameters	Optional	Express transaction date formatted YYYYMMDD
ExpressTransactionTime	Parameters	Optional	Express transaction time formatted HHMMSS
HostBatchID	Parameters	Optional	Unique host batch identifier
HostItemID	Parameters	Optional	Unique host batch item identifier
ReferenceNumber	Parameters	Optional	User-defined reference number
ShiftID	Parameters	Optional	Shift ID
TrackingID	Parameters	Optional	Internal transaction tracking identifier
TransactionAmount	Parameters	Optional	Transaction amount
TransactionSetupID	Parameters	Optional	TransactionSetupID
TransactionStatusCode	Parameters	Optional	Transaction status code
TransactionType	Parameters	Optional	Transaction type
ReverseOrder	Parameters	Optional	Flag to query records in descending order

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

Name	Class	Returned	Description
ReportingData	Response	Returned	Transaction information in XML format. A maximum of 1000 records can be returned.
ReportingID	Response	Returned	Reporting identifier

Services Methods

PaymentAccountCreate

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

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	Merchant ID
PaymentAccountType	PaymentAccount	Required	Payment Account Type
PaymentAccountReferenceNumber	PaymentAccount	Required	Payment Account Reference Number
PASSUpdaterBatchStatus	PaymentAccount	Optional [2]	Specifies whether or not the Payment Account will be included in the next PASS Updater batch
MagneprintData	Card	Conditional [1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional [1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional [1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional [1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional [1]	Track2 Data
Track1Data	Card	Conditional [1]	Track1 Data
CardNumber	Card	Conditional [1]	Card Number
ExpirationMonth	Card	Conditional [1]	Expiration Month
ExpirationYear	Card	Conditional [1]	Expiration Year
AccountNumber	DemandDepositAccount	Conditional [3]	Account Number of consumer bank account
RoutingNumber	DemandDepositAccount	Conditional [3]	Routing Number of consumer bank account
[Address Fields]	Address	Optional	See Address Class

Conditional [1] If PaymentAccountType = 0 (CreditCard), then 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

Optional [2] Used with Account Updater service only.

Conditional [3] Required if PaymentAccountType is a DDA 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	Services identifier
PaymentAccountID	Response.PaymentAccount	Returned	Unique GUID that identifies the Payment Account. The Express platform generates this.

Name	Class	Returned	Description
PaymentAccountReferenceNumber	Response.PaymentAccount	Returned	User-defined reference number to identify a Payment Account

PaymentAccountDelete

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

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	Merchant ID
PaymentAccountID	PaymentAccount	Required	Unique GUID that identifies the Payment

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	Services identifier

PaymentAccountUpdate

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

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	Merchant ID
PaymentAccountID	PaymentAccount	Required	Unique GUID that identifies the Payment
PaymentAccountType	PaymentAccount	Required[3]	Payment Account Type
PaymentAccountReferenceNumber	PaymentAccount	Required	Payment Account Reference Number
MagneprintData	Card	Conditional[1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional[1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional[1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional[1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional[1]	Track2 Data
Track1Data	Card	Conditional[1]	Track1 Data
CardNumber	Card	Conditional[1]	Card Number
ExpirationMonth	Card	Conditional[1]	Expiration Month
ExpirationYear	Card	Conditional[1]	Expiration Year
AccountNumber	DemandDepositAccount	Conditional[2]	Account Number of consumer bank account
RoutingNumber	DemandDepositAccount	Conditional[2]	Routing Number of consumer bank account

Name	Class	Required	Description
[Address Fields]	Address	Optional	See Address Class

Conditional[1] If PaymentAccountType = 0 (CreditCard), then 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

Conditional[2] Required if PaymentAccountType is a DDA account.

Required[3] PaymentAccountType cannot be updated.

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	Services identifier
PaymentAccountID	Response.PaymentAccount	Returned	Unique GUID that identifies the Payment Account. The Express platform generates this.
PaymentAccountReferenceNumber	Response.PaymentAccount	Returned	User-defined reference number to identify a Payment Account

PaymentAccountQuery

The PaymentAccountQuery method is used to query existing Payment Account (PASS) record(s).

- The QueryData field will return Payment Account information in XML format
- A maximum of 1000 records are returned

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	Merchant ID
PaymentAccountID	PaymentAccountParameters	Optional	Unique GUID that identifies the Payment
PaymentAccountReferenceNumber	PaymentAccountParameters	Optional	User-defined reference number to identify a Payment Account
PaymentBrand	PaymentAccountParameters	Optional	Card type query value
ExpirationMonthBegin	PaymentAccountParameters	Optional	2-digit month (e.g. 02)
ExpirationMonthEnd	PaymentAccountParameters	Optional	2-digit month (e.g. 02)
ExpirationYearBegin	PaymentAccountParameters	Optional	2-digit year (e.g. 25)
ExpirationYearEnd	PaymentAccountParameters	Optional	2-digit year (e.g. 25)
TransactionSetupID	PaymentAccountParameters	Optional	Unique GUID that identifies the TransactionSetupID
PASSUpdaterDateTimeBegin	PaymentAccountParameters	Optional	Begin date of range. Format "YYYYMMDD"
PASSUpdaterDateTimeEnd	PaymentAccountParameters	Optional	End date of range. Format "YYYYMMDD"
PASSUpdaterBatchStatus	PaymentAccountParameters	Optional	Specifies whether or not the Payment Account will be included in the next PASS Updater batch
PASSUpdaterStatus	PaymentAccountParameters	Optional	Specifies the match status of the PASS record

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message

Name	Class	Returned	Description
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	Services identifier
QueryData	Response	Returned	Query information in XML format
PaymentAccountID	Response.PaymentAccount	Conditional	Unique GUID that identifies the Payment Account. Returned if input value passed.
PaymentAccountReferenceNumber	Response.PaymentAccount	Conditional	User-defined reference number to identify a Payment Account. Returned if input value passed.

Additional notes on PaymentAccountQuery

- This method will return the resulting query in XML format based on the input parameters passed in via the PaymentAccountParameters node.
- At least one PaymentAccountParameters value is 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	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	Merchant ID
PaymentAccountID	PaymentAccount	Required	Unique GUID that identifies the Payment
PASSUpdaterBatchStatus	PaymentAccount	Required	Specifies whether or not the Payment Account will be included in the next PASS Updater batch
PASSUpdaterOption	PaymentAccount	Optional	Specifies whether or not the PASS record will be updated automatically

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	Services identifier
PaymentAccountID	Response.PaymentAccount	Returned	Unique GUID that identifies the Payment Account. The Express platform generates this.

PaymentAccountCreateWithTransID

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

Input Fields

Name	Class	Required	Description
ApplicationID	Application	Required	Unique application identifier

Name	Class	Required	Description
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	Merchant ID
PaymentAccountType	PaymentAccount	Required	Payment Account Type
PaymentAccountReferenceNumber	PaymentAccount	Required	Payment Account Reference Number
PASSUpdaterBatchStatus	PaymentAccount	Optional[1]	Specifies whether or not the Payment Account will be included in the next PASS Updater batch
TransactionID	Transaction	Required	TransactionID returned in original transaction
AccountNumber	DemandDepositAccount	Conditional[2]	Account Number of consumer bank account
RoutingNumber	DemandDepositAccount	Conditional[2]	Routing Number of consumer bank account
[Address Fields]	Address	Optional	See Address Class

Optional[1] Used with Account Updater service only.

Conditional[2] Required if PaymentAccountType is a DDA 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	Services identifier
PaymentAccountID	Response.PaymentAccount	Returned	Unique GUID that identifies the Payment Account. The Express platform generates this.
PaymentAccountReferenceNumber	Response.PaymentAccount	Returned	User-defined reference number to identify a Payment Account

Additional notes on PaymentAccountCreateWithTransID

- TransactionID's from the following Transaction Types may be used to generate a PaymentAccountID: CreditCardSale, CreditCardCredit, CreditCardAuthorization, CreditCardAVSOnly, CreditCardForce, DebitCardSale, DebitCardReturn, CheckSale, CheckCredit, CheckVerification
- The PaymentAccountType submitted must be correct and correspond for the associated TransactionID submitted.
- TransactionsIDs are maintained per the Worldpay data retention policy for 45 day. The PaymentAccountCreateWithTransID can only be used to create Payment Account records during this 45 day window.
- In order to store address details associated with the token on our platform, the PaymentAccountCreateWithTransID request must include the appropriate address details you wish to associate to the token. The address information from the original TransactionID WILL NOT be automatically stored. It must be explicitly submitted as part of the PaymentAccountCreateWithTransID request.

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	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	Merchant 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
ServicesID	Response	Returned	Services identifier
QueryData	Response	Returned	Query information in XML format. <QueryData><Results> … </Results> … </Results>. Within <Results> … </Results>: RecordCount = # PASS Records PageCount = # of Pages

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
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	Merchant ID
Page	Paging	Required	PASS Token Report page number (e.g. 1, 2, 3, …)

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	Services identifier
QueryData	Response	Returned	Query information in XML format. Records returned in xml format <Items><I> … </I> … </Items> Within <I> … </I>: <T>…</T> = PaymentAccountID <R>…</R> = PaymentAccount ReferenceNumber

ScheduledTaskDelete

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

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	Merchant ID
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
ServicesID	Response	Returned	Services identifier

ScheduledTaskQuery

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

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	Merchant ID
ScheduledTaskID	ScheduledTaskParameters	Optional	Unique value that identifies the Scheduled Task
ScheduledTaskReferenceNumber	ScheduledTaskParameters	Optional	User-defined reference number to identify a Scheduled Task
ScheduledTaskName	ScheduledTaskParameters	Optional	User-defined name of the task
ScheduledTaskGroupID	ScheduledTaskParameters	Optional	Group ID. Express platform generates this.
RunStartDate	ScheduledTaskParameters	Optional	Start date. Format YYYYMMDD
RunUntilCancelFlag	ScheduledTaskParameters	Optional	Use ExtendedBooleanType. If set to true, the task will run every scheduled time until the task is deleted by the user.
RunFrequency	ScheduledTaskParameters	Optional	Use ExtendedRunFrequency. Specifies how often an event should occur.
ScheduledTaskStatus	ScheduledTaskParameters	Optional	Use ExtendedStatusType. Status of the task. Can be set to Active or Disabled.
ScheduledTaskQueryType	ScheduledTaskParameters	Optional	Query type used to provide additional event information
ScheduledTaskRunStatus	ScheduledTaskParameters	Optional	Run status of the task
ReferenceNumber	Transaction	Optional	Reference Number of the originally-scheduled transaction

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	Services identifier
QueryData	Response	Returned	Query information in XML format. Max number of records returned by ScheduledTaskQuery increased from 1,000 to 2,500 in June 2021.
ScheduledTaskID	Response.ScheduledTask	Returned	Unique value that identifies the Scheduled Task
ScheduledTaskReferenceNumber	Response.ScheduledTask	Conditional	User-defined reference number to identify a Scheduled Task. Returned if input value passed.

Additional notes on ScheduledTaskQuery

- At least one ScheduledTaskParameters value is required to query records.

ScheduledTaskUpdate

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

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	Merchant ID
ScheduledTaskID	ScheduledTask	Required	Unique value that identifies the Scheduled Task
ScheduledTaskReferenceNumber	ScheduledTask	Required	User-defined reference number to identify a Scheduled Task
RunFrequency	ScheduledTask	Required	Number of times this task should run
RunUntilCancelFlag	ScheduledTask	Required	If set to true, the task will run every scheduled time until the task is deleted by the user.
RunCycles	ScheduledTask	Conditional	Number of times this task should run. Required only when RunUntilCancelFlag is set to False.
ScheduledTaskStatus	ScheduledTask	Required	Use ExtendedStatusType. Status of the task. Can be set to Active or Disabled.
ScheduledTaskName	ScheduledTask	Optional	User-defined name of the task
ScheduledTaskGroupID	ScheduledTask	Optional	Group ID. 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	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Services identifier

ScheduledTaskRetry

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

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	Merchant ID
ScheduledTaskRunLogID	ScheduledTask	Optional	Run log ID of a 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
ServicesID	Response	Returned	Services identifier
TransactionID	Response.Transaction	Returned	Transaction identifier
ProcessorName	Response.Transaction	Returned	Name of processor
TransactionStatus	Response.Transaction	Returned	Description of transaction status
TransactionStatusCode	Response.Transaction	Returned	Status of transaction
ScheduledTaskID	Response.ScheduledTask	Returned	Unique value that identifies the Scheduled Task

TokenCreate

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

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	Merchant ID
MagneprintData	Card	Conditional[1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional[1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional[1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional[1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional[1]	Track2 Data
Track1Data	Card	Conditional[1]	Track1 Data
CardNumber	Card	Conditional[1]	Card Number
ExpirationMonth	Card	Conditional[1]	Expiration Month
ExpirationYear	Card	Conditional[1]	Expiration Year
AccountType	CheckAccount	Conditional[2]	Account Type
AccountNumber	CheckAccount	Conditional[2]	Account Number of consumer bank account
RoutingNumber	CheckAccount	Conditional[2]	Routing Number of consumer bank account
TokenProvider	Token	Required	Token provider
VaultID	Token	Conditional	Merchant VaultID included only for Paymetric TokenProvider

Name	Class	Required	Description
TokenRegistrationID	Token	Conditional	Registration ID. Can be used to generate an OmniToken
TerminalSerialNumber	Terminal	Optional	Terminal Serial Number
LaneNumber	Terminal	Optional	Lane number. Recommend using 01 to 99 for support across all processors.

Conditional[1] If PaymentAccountType = 0 (CreditCard), then 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

Conditional[2] Submitted instead of Card information if generating token for ACH data

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	Services identifier
TruncatedCardNumber	Response.Card	Conditional	Truncated card number. Not returned with ACH token.
ExpirationMonth	Response.Card	Conditional	Expiration month. Not returned with ACH token.
ExpirationYear	Response.Card	Conditional	Expiration year. Not returned with ACH token.
CardLogo	Response.Card	Conditional	Payment brand. Not returned with ACH token.
TokenID	Response.Token	Returned	Token
TokenNewlyGenerated	Response.Token	Returned	Identifies whether token has been created in the past for the same card
TokenProvider	Response.Token	Returned	Token provider

TokenCreateWithTransID

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

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	Merchant ID
TransactionID	Transaction	Required	TransactionID returned in original transaction
AccountType	CheckAccount	Conditional[1]	Account Type
TokenProvider	Token	Required	Token provider
VaultID	Token	Conditional	Merchant VaultID included only for Paymetric TokenProvider
TerminalSerialNumber	Terminal	Optional	Terminal Serial Number
LaneNumber	Terminal	Optional	Lane number. Recommend using 01 to 99 for support across all processors.

Conditional[1] Submitted only if generating token for ACH data

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

Name	Class	Returned	Description
ExpressTransactionTime	Response	Returned	Express transaction time formatted HHMMSS
ExpressTransactionTimezone	Response	Returned	Express transaction UTC time zone
ServicesID	Response	Returned	Services identifier
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
TokenNewlyGenerated	Response.Token	Returned	Identifies whether token has been created in the past for the same card
TokenProvider	Response.Token	Returned	Token provider

TokenRegistration

The TokenRegistration method is used to get a RegistrationID for a given card. Also used for mobile In App.

- The registration ID expires in 24 hours.
- This is only for OmniToken.
- Supported via XML only.

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	Merchant ID
MagneprintData	Card	Conditional[1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional[1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional[1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional[1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional[1]	Track2 Data
Track1Data	Card	Conditional[1]	Track1 Data
CardNumber	Card	Conditional[1]	Card Number. For mobile In-App, submit DPAN in this field.
ExpirationMonth	Card	Conditional[1]	Expiration Month
ExpirationYear	Card	Conditional[1]	Expiration Year
Cryptogram	Card	Optional[2]	Cryptogram for In-App
WalletType	Card	Optional[2]	Wallet type for In-App
ElectronicCommerceIndicator	Card	Optional[2]	Ecommerce indicator for In-App
TokenProvider	Token	Required	Token provider
TokenOptions	Token	Optional	If requesting a TokenRegistrationID based on a pior TokenID (OmniToken), set TokenOptions to a value of 1. Else do not submit TokenOptions.
TokenID	Token	Optional	TokenID (OmniToken) can be used as an input field to generate a TokenRegistrationID. If not using TokenID for this purpose, do not submit TokenID field.
TerminalSerialNumber	Terminal	Optional	Terminal Serial Number
LaneNumber	Terminal	Optional	Lane number. Recommend using 01 to 99 for support across all processors.

Conditional[1] If PaymentAccountType = 0 (CreditCard), then 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

Optional[2] For a mobile In App payment, the DPAN is passed in the CardNumber field and the Cryptogram is required. The WalletType and ElectronicCommerceIndicator are optional.

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	Services identifier
TruncatedCardNumber	Response.Card	Returned	Truncated card number
CardLogo	Response.Card	Returned	Payment brand
TokenRegistrationID	Response.Token	Returned	Registration ID
TokenID	Response.Token	Returned	Token
TokenProvider	Response.Token	Returned	Token provider

System Methods

HealthCheck

The HealthCheck method is used to 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	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	Merchant ID

Output Fields

Name	Class	Returned	Description
ExpressResponseCode	Response	Returned	Express Response Code
ExpressResponseMessage	Response	Returned	Express Response Message
HostResponseCode	Response	Conditional	Host Response Code.
HostResponseMessage	Response	Conditional	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

AccountTokenCreate

The AccountTokenCreate method is used to generate a new AccountToken that will replace an existing active AccountToken.

- The old AccountToken remains active until the new AccountToken is activated by calling AccountTokenActivate.
- Multiple calls to AccountTokenCreate without calling AccountTokenActivate will return the same new AccountToken as the first call to AccountTokenCreate. This ensures both that the old AccountToken

remains active until activated; and that the new AccountToken cannot be replaced between the time it is created and the time it is activated.

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	Merchant 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
NewAccountToken	Response.Credentials	Returned	New AccountToken
ProcessorName	Response.Transaction	Returned	Name of processor

AccountTokenActivate

The AccountTokenActivate method is used to activate the new AccountToken that was generated when calling AccountTokenCreate

- AccountTokenCreate will disable the previous AccountToken

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	Merchant ID
NewAccountToken	Credentials	Required	New AccountToken

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	Response.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	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	Merchant ID
TransactionAmount	Transaction	Required	Transaction amount
ReferenceNumber	Transaction	Required	Reference number
TicketNumber	Transaction	Required	Ticket number
MarketCode	Transaction	Required	Type of industry
PartialApprovedFlag	Transaction	Required	Specifies if POS supports partial approvals
DuplicateCheckDisableFlag	Transaction	Optional	Specifies whether duplicate checking is enabled on the transaction
DuplicateOverrideFlag	Transaction	Optional	Typically used during a transaction resubmission to override a previous duplicate decline
MerchantSuppliedTransactionID	Transaction	Optional	User-defined identifier
PaymentType	Transaction	Optional	Payment type used for credential on file transactions
SubmissionType	Transaction	Optional	Submission type used for credential on file transactions
NetworkTransactionID	Transaction	Optional	Transaction identifier used for credential on file transactions
ApprovalNumber	Transaction	Conditional	Approval Number. Required only when using TransactionSetupMethod 4 (CreditCardForce)
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Required	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
[Required Terminal Fields]	Terminal	Required	See Required Terminal Settings documentation from Developer Integrations.
[Other Optional Terminal Fields]	Terminal	Optional	See Terminal Class
PaymentAccountID	PaymentAccount	Conditional	Payment Account Identifier. Needed only for updating a Payment Account.
PaymentAccountType	PaymentAccount	Conditional	Payment Account Type. Needed only for creating or updating a Payment Account.
PaymentAccountReferenceNumber	PaymentAccount	Conditional	Payment Account Reference Number. Needed only for creating or updating a Payment Account.
AddressEditAllowed	Address	Optional	Specifies whether cardholder can edit address directly on Hosted Payments page
[Address Fields]	Address	Optional	See Address Class. For keyed transactions, BillingAddress1 and BillingZipcode should be included.
TransactionSetupMethod	TransactionSetup	Required	Transacton method
Device	TransactionSetup	Optional	Entry device used
Embedded	TransactionSetup	Required	Specifies if card entry page will be embedded in an application
CVVRequired	TransactionSetup	Optional	Specifies if user must enter CVV2 / CVC2 / CID values
AutoReturn	TransactionSetup	Optional	Specifies if approval detail will be auto-redirected to ReturnURL
DeviceInputCode	TransactionSetup	Optional	Device input code
CompanyName	TransactionSetup	Optional[1]	Name of merchant
LogoURL	TransactionSetup	Optional[1]	HTTPS URL of merchant logo to be displayed when Embedded flag is set to false
Tagline	TransactionSetup	Optional[1]	Text displayed next to logo
WelcomeMessage	TransactionSetup	Optional[1]	Text to display to cardholder (if not provided, a generic message will be displayed)
ReturnURL	TransactionSetup	Optional	Return URL link displayed after transaction completion or cancellation

Name	Class	Required	Description
ReturnURLTitle	TransactionSetup	Optional	Text to display for ReturnURL (if not provided, a generic "return to merchant" message will be displayed)
OrderDetails	TransactionSetup	Optional[1]	Text to describe purchase
ProcessTransactionTitle	TransactionSetup	Optional	Text used for submit button label
CustomCss	TransactionSetup	Optional	Custom CSS to style the payment page. See Custom CSS appendix.
[Healthcare Fields]	ExtParm.Healthcare	Optional	See Healthcare Class. May be required for FSA / HSA cards.
[Auto Rental Fields]	ExtParm.AutoRental	Optional	See AutoRental Class. For Auto Rental transactions, only AutoRentalDuration field is supported.
[Lodging Fields]	ExtParm.Lodging	Optional	See Lodging Class. For Lodging transactions all fields are required.
[Token Fields]	ExtParm.Token	Optional	See Token Class. TokenProvider only be used for Hosted Payments Premium integrations to request an OmniToken in the redirect response. Contact Developer Integrations for more information.

Optional[1] Only used when Embedded flag is set to 0/False.

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	TransactionSetupID
TransactionSetupID	Response.PaymentAccount	Returned	TransactionSetupID
TransactionSetupID	Response.TransactionSetup	Returned	TransactionSetupID
ValidationCode	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
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	Merchant ID
TransactionSetupID	TransactionSetup	Required	TransactionSetupID

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. The input fields are the same for both methods; the output fields are different.

1. The BINQuery method is maintained for existing applications.
2. The EnhancedBINQuery method provides the most detailed BIN information and is recommended for all new development.

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
ApplicationVersion	Application	Required	Version of application
AccountID	Credentials	Required	Unique account identifier
AccountToken	Credentials	Required	Secret token used for authentication
AcceptorID	Credentials	Required	Merchant ID
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
MagneprintData	Card	Conditional[1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional[1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional[1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional[1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional[1]	Track2 Data
Track1Data	Card	Conditional[1]	Track1 Data
CardNumber	Card	Conditional[1]	Card Number
ExpirationMonth	Card	Conditional[1]	Expiration Month
ExpirationYear	Card	Conditional[1]	Expiration Year
PaymentAccountID	ExtParm.PaymentAccount	Conditional[1]	Payment Account Identifier

Conditional[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.

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
BINTypeCode	Response.BIN	Returned	Numeric BIN type code
BINTypeValue	Response.BIN	Returned	Text associated with the numeric BIN type code
BINDecorator	Response.BIN	Returned	Unique value defined by merchant or vendor and maps to BIN type value

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.

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

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	Merchant ID
TerminalID	Terminal	Required	Terminal identifier
LaneNumber	Terminal	Optional	Lane Number
TerminalSerialNumber	Terminal	Optional	Terminal serial number (encrypted devices)
MagneprintData	Card	Conditional[1]	Magneprint Data
EncryptedTrack2Data	Card	Conditional[1]	Encrypted Track2 Data
EncryptedTrack1Data	Card	Conditional[1]	Encrypted Track1 Data
EncryptedCardData	Card	Conditional[1]	Encrypted Card Data
CardDataKeySerialNumber	Card	Conditional	Card encryption KSN. May be required with encrypted data.
EncryptedFormat	Card	Conditional	Card encryption format. May be required with encrypted data.
Track2Data	Card	Conditional[1]	Track2 Data
Track1Data	Card	Conditional[1]	Track1 Data
CardNumber	Card	Conditional[1]	Card Number
ExpirationMonth	Card	Conditional[1]	Expiration Month
ExpirationYear	Card	Conditional[1]	Expiration Year
TransactionID	Transaction	Conditional[1]	Prior TransactionID (within past 45 days) where card data, a PASS Token, or OmniToken was used
PaymentAccountID	ExtParm.PaymentAccount	Conditional[1]	Payment Account Identifier

Conditional[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 **or** EncrypedEMVData **or** TransactionID.

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
Status	Response.EnhancedBIN	Returned	Query status
CheckCard	Response.EnhancedBIN	Returned	Check Card indicator
CommercialCard	Response.EnhancedBIN	Returned	Commercial Card indicator
CreditCard	Response.EnhancedBIN	Returned	Credit Card indicator
DebitCard	Response.EnhancedBIN	Returned	Debit Card indicator
DurbinBINRegulation	Response.EnhancedBIN	Returned	Durbin BIN Regulation indicator
EBT	Response.EnhancedBIN	Returned	EBT indicator
FleetCard	Response.EnhancedBIN	Returned	Fleet Card indicator
GiftCard	Response.EnhancedBIN	Returned	Gift Card indicator
HSAFSACard	Response.EnhancedBIN	Returned	HSA / FSA Card indicator
InternationalBIN	Response.EnhancedBIN	Returned	International BIN indicator
PinLessBillPay	Response.EnhancedBIN	Returned	Pinless Bill Pay indicator
PrepaidCard	Response.EnhancedBIN	Returned	Prepaid Card indicator
WIC	Response.EnhancedBIN	Returned	WIC indicator

Classes

This section describes the list of Classes used with Express.

Address Class

Name	Type	Max	Description
BillingName	String	100	The name used for billing purposes
BillingEmail	String	80	The email address used for billing purposes
BillingPhone	String	20	The phone number used for billing purposes
BillingAddress1	String	50	The street address used for billing purposes
BillingAddress2	String	50	The street address used for billing purposes
BillingCity	String	40	The name of the city used for billing purposes
BillingState	String	30	The name of the state used for billing purposes
BillingZipcode	String	20	The zip code used for billing purposes
ShippingName	String	100	The name used for shipping purposes
ShippingEmail	String	80	The email address used for shipping purposes
ShippingPhone	String	20	The phone number used for shipping purposes
ShippingAddress1	String	50	The street address used for shipping purposes
ShippingAddress2	String	50	The street address used for shipping purposes
ShippingCity	String	40	The name of the city used for shipping purposes
ShippingState	String	30	The name of the state used for shipping purposes
ShippingZipcode	String	20	The zip code used for shipping purposes
AddressEditAllowed	String	1	Only used with XML and only when Embedded=0/False. Allows user to update address directly on Hosted Payments page. 0 = Cardholder cannot edit address on Hosted Payments page 1 = Cardholder can edit address on Hosted Payments page

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.

Name	Type	Max	Description
ApplicationID	String	40	Unique application identifier
ApplicationName	String	50	Name of application
ApplicationVersion	String	50	Version of application
HostApplicationID	String	4	TSYS-assigned Application ID. For use with TSYS direct EMV certifications only. XML only.
HostDeveloperID	String	6	TSYS-assigned Developer ID. For use with TSYS direct EMV certifications only. XML only.

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 Vantiv, TSYS, or FDC platform only.

Name	Type	Max	Description
AutoRentalAgreementNumber	String	14	Auto Rental agreement number
AutoRentalNoShowIndicator	String	1	Indicates whether rental vehicle was reserved, but not actually rented. 0 = Not Applicable 1 = No Show

Name	Type	Max	Description
AutoRentalExtraChargesDetail	String	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
AutoRentalPickupDate	String	8	Auto Rental pickup date. Format is YYYYMMDD.
AutoRentalDropoffDate	String	8	Reserved for future use
AutoRentalCustomerName	String	20	Name of the person or business with the reservation
AutoRentalReturnCity	String	18	City where vehicle was returned
AutoRentalReturnState	String	3	State where vehicle was returned
AutoRentalReturnLocationID	String	10	Auto Rental location ID where vehicle was returned
AutoRentalDuration	String	2	Number of rental days (e.g. 03). Pass 01 for no-show rentals.
AutoRentalPickupLocation	String	38	Name of business where vehicle was picked up
AutoRentalPickupCity	String	18	Name of city, town, or village where vehicle was picked up
AutoRentalPickupState	String	3	Abbreviation of state, province, or other country subdivision where vehicle was picked up
AutoRentalPickupCountryCode	String	3	Country code where vehicle was picked up. For US, use 840.
AutoRentalPickupTime	String	6	Local time when vehicle was picked up. Format HHMMSS.
AutoRentalReturnCountryCode	String	3	Country code where vehicle was returned. For US, use 840.
AutoRentalReturnDate	String	8	Date when vehicle was returned. Format YYYYMMDD.
AutoRentalReturnTime	String	6	Local time when vehicle was returned. Format HHMMSS.
[AutoRentalVehicleClassCode](#)	Enumeration	2	Classification of rental vehicle
AutoRentalDistance	String	5	Numeric distance traveled during rental period
[AutoRentalDistanceUnit](#)	Enumeration	2	Unit of measurement for distance traveled
[AutoRentalAuditAdjustmentCode](#)	Enumeration	2	Indicates if adjustment for mileage, fuel, vehicle damage, etc. was made to a rental agreement
AutoRentalAuditAdjustmentAmount	String	12	Adjustment amount for mileage, fuel, vehicle damage, etc. after a vehicle was returned
AutoRentalDailyRentalRate	String	10	Rate charged per day. Either daily or weekly rate should be included in standard amount format.
AutoRentalWeeklyRentalRate	String	10	Rate charged per week. Either daily or weekly rate should be included in standard amount format.
AutoRentalInsuranceCharges	String	10	Insurance amount (if charged). If purchased, should be amount greater than 0.00.

Batch Class

Name	Type	Max	Description
HostBatchID	String	10	Unique host batch identifier. HostBatchID values may be reused, though not more than once every 9999 batches per merchant. Note: At one merchant batch submitted per day, it would take a merchant over 27 years to reach this threshold.
HostItemID	String	25	Unique host batch item identifier
[BatchCloseType](#)	Enumeration	2	Type of batch close
HostBatchItems	String	N/A	
[BatchQueryType](#)	Enumeration	2	Batch query type
HostBatchCount	String	5	Batch count
HostBatchAmount	String	12	Batch amount
HostCreditSaleCount	String	5	Total numeric count of credit sales
HostCreditSaleAmount	String	12	Total dollar amount of credit sales with no decimal place: example $21.52 would be 2152
HostCreditReturnCount	String	5	Total numeric count of credit returns
HostCreditReturnAmount	String	12	Total dollar amount of credit returns with no decimal place: example $21.52 would be 2152
HostDebitSaleCount	String	5	Total numeric count of debit sales

Name	Type	Max	Description
HostDebitSaleAmount	String	12	Total dollar amount of debit sales with no decimal place: example $21.52 would be 2152
HostDebitReturnCount	String	5	Total numeric count of debit returns
HostDebitReturnAmount	String	12	Total dollar amount of credit returns with no decimal place: example $21.52 would be 2152
BatchGroupingCode	Enumeration	2	See enum description
BatchIndexCode	Enumeration	2	See enum description
HostReversalQueueID	String	25	Reversal queue ID

BIN Class

The BIN class is used when performing BINQuery requests.

Name	Type	Max	Description
BINTypeCode	String	4	Numeric BIN type code 0 = NotFound 1 = Debit 2 = Healthcare 3 = Commercial 4 = Other 5 = PrepaidDebit
BINTypeValue	String	75	Text associated with the numeric BIN type code
BINDecorator	String	75	Unique value defined by merchant or vendor and maps to BIN type value. Returned if BIN is found.

Card Class

The Card Class is used to pass cardholder information.

Name	Type	Max	Description
CardNumber	String	30	Numeric number displayed on front of card
ExpirationMonth	String	2	2-digit expiration month displayed on front of card, e.g. 02. May also be returned in the response.
ExpirationYear	String	2	2-digit expiration year displayed on front of card, e.g. 25. May also be returned in the response.
CardholderName	String	80	Name of the cardholder as printed on the front of card
Track1Data	String	76	Track1 data embedded on the magnetic strip on the back of card
Track2Data	String	37	Track2 data embedded on the magnetic strip on the back of card
Track3Data	String	------	Reserved for future use
MagneprintData	String	1500	Encrypted track data sent from an encrypted swipe reader. Do not parse, as encrypted format may vary.
PINBlock	String	80	Encrypted PIN block information sent from a PIN encryption device
AVSResponseCode	String	2	AVS response from the authorization request
CVV	String	4	Card Verification Value found on the card (CVV2, CVC2, CID)
CVVResponseCode	String	2	CVV response from the authorization request
CAVV	String	60	Cardholder Authentication Verification Value (future use)
CAVVResponseCode	String	10	CAVV response from the authorization request (future use)
XID	String	60	Verified by Visa value (future use)
KeySerialNumber	String	20	Debit Key Serial Number
CardLogo	String	20	Payment brand. Possible values include: "Visa", "Mastercard", "Discover", "Amex", "Diners Club", "JCB", "Carte Blanche", "Other", "Union Pay"
EncryptedTrack1Data	String	300	Encrypted Track1 Data
EncryptedTrack2Data	String	200	Encrypted Track2 Data
EncryptedCardData	String	200	Encrypted Card Data (keyed)
CardDataKeySerialNumber	String	26	Used when any encrypted track or card fields are used

Name	Type	Max	Description
EncryptedFormat	Enumeration	2	Encrypted Format
TruncatedCardNumber	String	30	Truncated card number. Format xxxx-xxxx-xxxx-1234
CardNumberMasked	String	30	Truncated card number. Format xxxx-xxxx-xxxx-1234
Cryptogram	String	60	Base64-encoded Cryptogram associated with an In-App DPAN or 3-D Secure 2.0 transaction
WalletType	Enumeration	1	Wallet Type of the mobile In-App device
ElectronicCommerceIndicator	String	2	Electronic Commerce Indicator
BIN	String	6	First six digits of card number. XML only.
CardLevelResults	String	6	Card product identifier. XML only (Vantiv).

Additional information on Card Class

- 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.

CheckAccount Class

The CheckAccount class is used when generating OmniToken values for ACH data.

Name	Type	Max	Description
AccountType	Enumeration	2	ACH account type
AccountNumber	String	20	ACH Account Number
RoutingNumber	String	9	ACH Routing Number

Credentials Class

The Credentials Class is required for every transaction and is used to authenticate the sender.

AccountID and AccountToken are assigned by Worldpay Integrated Payments.

Name	Type	Max	Description
AccountID	String	10	Unique account identifier
AccountToken	String	140	Secret token used for authentication
AcceptorID	String	50	Unique merchant account identifier, most likely a Merchant ID
NewAccountToken	String	140	New secret token used for authentication

DemandDepositAccount Class

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

Name	Type	Max	Description
DDAAccountType	Enumeration	2	Type of DDA account. Checking, Savings, etc. 0 = Checking 1 = Savings
AccountNumber	String	20	Account Number
RoutingNumber	String	9	Routing Number of the financial institution
CheckNumber	String	8	Check Number
CheckType	Enumeration	2	Type of check. Business, Personal, etc. 0 = Personal 1 = Business
TrunxatedAccountNumber	String	20	Masked return value. Used for display purposes.
TruncatedRoutingNumber	String	9	Masked return value. Used for display purposes.

EnhancedBIN Class

This class is used for returning BIN details for the EnhancedBINQuery method.

Name	Type	Max	Description
Status	String	20	Status of the query. Returns "Found" or "NotFound".
CheckCard	String	1	Check Card indicator. Returns "Y" if applicable to queried card.
CommercialCard	String	1	Commercial Card indicator. Returns "Y" if applicable to queried card.
CreditCard	String	1	Credit Card indicator. Returns "Y" if applicable to queried card.
DebitCard	String	1	Debit Card indicator. Returns "Y" if applicable to queried card.
DurbinBINRegulation	String	1	Durbin BIN Regulation indicator. If present, returns integer indicating the type of Durbin regulation applicable to queried card.
EBT	String	1	EBT indicator. Returns "Y" if applicable to queried card.
FleetCard	String	1	Fleet Card indicator. Returns "Y" if applicable to queried card.
GiftCard	String	1	Gift Card indicator. Returns "Y" if applicable to queried card.
HSAFSACard	String	1	HSA / FSA Card indicator. Returns "Y" if applicable to queried card.
InternationalBIN	String	1	International BIN indicator. Returns "Y" if applicable to queried card.
PinLessBillPay	String	1	Pinless Bill Pay indicator. Returns "Y" if applicable to queried card.
PrepaidCard	String	1	Prepaid Card indicator. Returns "Y" if applicable to queried card.
WIC	String	1	WIC indicator. Returns "Y" if applicable to queried card.

ExtendedParameters Class

Name	Type	Max	Description
Key	String		Name of Key
Value	String		Value of Key

Note: 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. Note that an amount type of 2/Healthcare is required when including Healthcare/IIAS details. Please note that Healthcare fields are only required for merchants/integrations that wish to accept FSA/HSA cards at merchant locations that also sell non-healthcare goods (such as with pharmacies and big-box retailers). Healthcare fields are not used to accept FSA/HSA cards for merchant locations that provide only healthcare goods and services (such as doctors offices and hospitals).

Name	Type	Max	Description
HealthcareFlag	BooleanType	1	If this is set to True, the Healthcare parameters should be submitted.
HealthcareFirstAccountType	Enumeration	2	Account Type
HealthcareFirstAmountType	Enumeration	2	Amount Type
HealthcareFirstCurrencyCode	String	3	Currency Code. Use 840 for U.S.
HealthcareFirstAmountSign	Enumeration	2	0 = Positive amount 1 = Negative amount
HealthcareFirstAmount	String	12	Healthcare Amount
HealthcareSecondAccountType	Enumeration	2	Account Type
HealthcareSecondAmountType	Enumeration	2	Amount Type
HealthcareSecondCurrencyCode	String	3	Currency Code. Use 840 for U.S.
HealthcareSecondAmountSign	Enumeration	2	0 = Positive amount 1 = Negative amount
HealthcareSecondAmount	String	12	Healthcare Amount

Name	Type	Max	Description
HealthcareThirdAccountType	Enumeration	2	Account Type
HealthcareThirdAmountType	Enumeration	2	Amount Type
HealthcareThirdCurrencyCode	String	3	Currency Code. Use 840 for U.S.
HealthcareThirdAmountSign	Enumeration	2	0 = Positive amount 1 = Negative amount
HealthcareThirdAmount	String	12	Healthcare Amount
HealthcareFourthAccountType	Enumeration	2	Account Type
HealthcareFourthAmountType	Enumeration	2	Amount Type
HealthcareFourthCurrencyCode	String	3	Currency Code. Use 840 for U.S.
HealthcareFourthAmountSign	Enumeration	2	0 = Positive amount 1 = Negative amount
HealthcareFourthAmount	String	12	Healthcare Amount

Identification Class

Name	Type	Max	Description
BirthDate	String	8	Date of Birth. Format YYYYMMDD
DriversLicenseNumber	String	40	Drivers License Number
DriversLicenseState	String	2	2-character state abbreviation (e.g. CA, NY, AZ) where the license was issued
TaxIDNumber	String	9	SSN or Tax ID (no dashes)

Image Class

Name	Type	Max	Description
FileName	String	250	File Name of the Check image
ImageData	Byte Array	512 kb	Check image (front or back)
ImageSide	String	16	Check image side. Allowed values are "Front" or "Back".

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 Vantiv, TSYS or FDC platform only.

Name	Type	Max	Description
LodgingAgreementNumber	String	25	Hotel Folio number
LodgingCheckInDate	String	8	Check-in date. Format YYYYMMDD Advance check-in = date of reservation No-show = anticipated arrival date
LodgingCheckoutDate	String	8	Check-out date. Format YYYYMMDD
LodgingRoomAmount	String	12	Nightly rate for one room. Must always be greater than 0.00 for room reservation
LodgingRoomTax	String	12	Reserved for future use
LodgingNoShowIndicator	String	1	Indicates a room was reserved, but not actually rented 0 = Not Applicable 1 = No Show
LodgingDuration	String	2	Number of hotel stay days (e.g. 03). Pass 01 for no-show. From 01 to 99.
LodgingCustomerName	String	26	Lodging customer name
LodgingClientCode	String	25	Reserved for future use
LodgingExtraChargesDetail	String	6	Indicates any ancillary charges. Field can contain multiple indicators 0 = Not Used 1 = Reserved 2 = Restaurant

Name	Type	Max	Description
			3 = Gift Shop
			4 = Mini-Bar
			5 = Telephone
			6 = Other
			7 = Laundry
LodgingExtraChargesAmounts	String	250	Reserved for future use
LodgingPrestigiousPropertyCode	Enumeration	2	Prestigious property code
LodgingSpecialProgramCode	Enumeration	2	Special program code
LodgingChargeType	Enumeration	2	Lodging charge type

Paging Class

The Paging Class is optional with the PaymentAccountQueryTokenReport method.

Name	Type	Max	Description
Page	String	9	PASS Token Report page number (e.g. 1, 2, 3, …)

Parameters Class

Used for reporting on transactions.

Name	Type	Max	Description
TransactionDateTimeBegin	String	30	Begin date/time of transaction range. Format "YYYY-MM-DD HH:MM:SS"
TransactionDateTimeEnd	String	30	End date/time of transaction range. Format "YYYY-MM-DD HH:MM:SS"
TransactionID	String	19	Transaction identifier
TerminalID	String	40	Terminal identifier
ApplicationID	String	40	Application identifier
ApprovalNumber	String	30	Issuer-assigned approval number
ApprovedAmount	String	10	Approved transaction amount
ExpressTransactionDate	String	30	Express transaction date. Format YYYYMMDD
ExpressTransactionTime	String	30	Express transaction time. Format HHMMSS
HostBatchID	String	10	Host batch identifier
HostItemID	String	25	Host batch item identifier
HostReversalQueueID	String	25	Reversal queue identifier
OriginalAuthorizedAmount	String	10	Original dollar amount authorized
ReferenceNumber	String	50	User-defined reference number
ShiftID	String	10	Shift identifier
SourceTransactionID	String	19	Transaction identifier from a previous transaction
TerminalType	String	2	Type of terminal
TrackingID	String	60	Internal transaction tracking identifier
TransactionAmount	String	12	Dollar amount of transaction
TransactionSetupID	String	50	Unique GUID that identifies the TransactionSetupID. The Express platform generates this.
TransactionStatus	String	60	Status of transaction
TransactionType	String	60	Type of transaction
XID	String	60	Verified by Visa value (future use)
ReverseOrder	String	2	Flag to query records in descending order 0 = Results returned in ascending order 1 = Results returned in descending order

PaymentAccount Class

Used in transactions utilizing our TransForm Tokenization/PASS service.

Name	Type	Max	Description
PaymentAccountID	String	50	Unique GUID that identifies the Payment Account. The Express platform generates this.
PaymentAccountType	Enumeration	2	Type of Payment Account
PaymentAccountReferenceNumber	String	50	User-defined reference number to identify a Payment Account
TransactionSetupID	String	50	Unique GUID that identifies the TransactionSetupID. The Express platform generates this.
PASSUpdaterBatchStatus	Enumeration	2	Specifies whether or not the Payment Account will be included in the next PASS Updater batch
PASSUpdaterOption	Enumeration	2	Specifies whether or not the PASS record will be updated automatically

PaymentAccountParameters Class

Used in transactions utilizing our TransForm Tokenization/PASS service.

Name	Type	Max	Description
PaymentAccountID	String	50	Unique GUID that identifies the Payment Account. The Express platform generates this.
PaymentAccountType	Enumeration	2	Type of Payment Account
PaymentAccountReferenceNumber	String	50	User-defined reference number to identify a Payment Account
PaymentBrand	String	50	Card type query value. Must match exactly, and can be: "Visa", "Mastercard", "Discover", "Amex", "Diners Club", "JCB", "Carte Blanche", "Other", "Union Pay"
ExpirationMonthBegin	String	2	2-digit month (e.g. 02)
ExpirationMonthEnd	String	2	2-digit month (e.g. 02)
ExpirationYearBegin	String	2	2-digit year (e.g. 25)
ExpirationYearEnd	String	2	2-digit year (e.g. 25)
TransactionSetupID	String	50	Unique GUID that identifies the TransactionSetupID. The Express platform generates this.
PASSUpdaterDateTimeBegin	String	30	Begin date of range. Format "YYYYMMDD"
PASSUpdaterDateTimeEnd	String	30	End date of range. Format "YYYYMMDD"
PASSUpdaterBatchStatus	Enumeration	2	Specifies whether or not the Payment Account will be included in the next PASS Updater batch
PASSUpdaterOption	Enumeration	2	Specifies whether or not the PASS record will be updated automatically
PASSUpdaterStatus	Enumeration	2	Specifies the match status of the PASS record

Response Class

The Response Class is used to communicate 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	Type	Max	Description
Credentials	Node	N/A	Any valid Credentials object. Used to authenticate the sender.
Card	Node	N/A	Any valid Card object. Use to pass cardholder information.
Batch	Node	N/A	Any valid Batch object. Use to pass batch information.
Transaction	Node	N/A	Any valid Transaction object. Used to pass transaction information.
ExpressResponseCode	String	10	Express Response Code
ExpressResponseMessage	String	100	Express Response Message
HostResponseCode	String	3	Host Response Code
HostResponseMessage	String	150	Host Response Message. Note that its return depends on the processor being used.
ExpressTransactionDate	String	8	Express platform transaction date
ExpressTransactionTime	String	8	Express platform transaction time
ExpressTransactionTimezone	String	50	Express platform transaction time zone
HostTransactionDateTime	String	50	Host platform transaction date and time

Name	Type	Max	Description
QueryData	String	N/A	Query information in XML format. A maximum of 1000 records can be returned. Format <items><item>…</item>…</items>
ReportingData	String	N/A	Transaction information in XML format
ServicesID	String	19	Services identifier
ReportingID	String	19	Reporting identifier

ScheduledTask Class

The ScheduledTask class is used to schedule transactions.

Name	Type	Max	Description
ScheduledTaskID	String	50	Unique value that identifies the Scheduled Task. The Express platform generates this.
ScheduledTaskReferenceNumber	String	50	User-defined reference number used to identify a Scheduled Task
RunFrequency	Enumeration	2	Specifies how often an event should occur
RunStartDate	String	8	The start date. Format YYYYMMDD
RunUntilCancelFlag	BooleanType	1	If set to true, the task will run every scheduled time until the task is deleted by the user
RunCycles	String	4	Number of times this task should run. Must be used when RunUntilCancelFlag is 0 (False).
ScheduledTaskStatus	Enumeration	2	Status of the task. Can be set to Active or Disabled.
ScheduledTaskName	String	50	User-defined name of the task
ScheduledTaskGroupID	String	50	Group ID. Express platform generates this.
ScheduledTaskRunLogID	String	50	Run Log ID of the task. Used with ScheduledTaskRetry and in the response of ScheduledTaskQuery where query type is 1 (RunLog).

ScheduledTaskParameters Class

The ScheduledTaskParameters Class is used in transactions utilizing our Scheduled Task service.

Name	Type	Max	Description
ScheduledTaskID	String	50	Unique value that identifies the Scheduled Task. The Express platform generates this.
ScheduledTaskReferenceNumber	String	50	User-defined reference number used to identify a Scheduled Task
ScheduledTaskQueryType	String	2	Query type used to provide additional event information 0 = Default 1 = RunLog
RunStartDate	String	8	Start date. Format YYYYMMDD
RunUntilCancelFlag	Enumeration	2	Use ExtendedBooleanType. If set to true, the task will run every scheduled time until the task is deleted by the user
RunFrequency	Enumeration	2	Use ExtendedRunFrequency. Specifies how often an event should occur.
ScheduledTaskStatus	Enumeration	2	Use ExtendedStatusType. Status of the task. Can be set to Active or Disabled.
ScheduledTaskRunStatus	Enumeration	2	Run status of the task
ScheduledTaskName	String	50	User-defined name of the task
ScheduledTaskGroupID	String	50	Group ID. Express platform generates this.

Terminal Class

Name	Type	Max	Description
TerminalID	String	40	Terminal identifier assign by Express
TerminalType	Enumeration	2	Type of terminal
TerminalCapabilityCode	Enumeration	2	Terminal capability
TerminalEnvironmentCode	Enumeration	2	Terminal environment
CardPresentCode	Enumeration	2	Card present indicator
CVVPresenceCode	Enumeration	2	CVV presence
CVVResponseType	Enumeration	2	CVV response type

Name	Type	Max	Description
CardInputCode	Enumeration	2	Card entry mode
CardholderPresentCode	Enumeration	2	Cardholder present indicator
MotoECICode	Enumeration	2	MOTO / Ecommerce indicator
ConsentCode	Enumeration	2	Consent code
TerminalEncryptionFormat	Enumeration	2	Reserved for future use
TerminalSerialNumber	String	40	Terminal Serial Number
TerminalModel	String	16	Terminal Model
LaneNumber	String	6	Lane number. Recommend using 01 to 99 for support across all processors.

Token Class

Name	Type	Max	Description
TokenID	String	100	Unique token representing the card
TokenProvider	Enumeration	2	Token provider 0 = Null 1 = ExpressPASS 2 = OmniToken 3 = Paymetric 4 = TransArmor (FDC only)
TokenNewlyGenerated	String	1	Indicates whether this is the first time a token has been created for the card under the merchant credentials 0 = Token has been generated in the past for the card 1 = Token has been generated for the first time for the card
VaultID	String	10	Merchant Vault ID. Response parameter only (except when generating Paymetric token).
TAProviderID	String	3	TransArmor Provider ID
TokenRegistrationID	String	19	Registration ID representing the card. Can also be used in a sale to process a transaction. Note that when TokenRegistrationID is obtained as part of an In-App transaction via eProtect, the Cryptogram, ExpirationMonth, and ExpirationYear may not be included in the request submitted to Express.
TokenOptions	String	1	Set TokenOptions to 1 only if using a TokenID (OmniToken) to request a TokenRegistrationID. Else TokenOptions should not be submitted.

Transaction Class

Name	Type	Max	Description
TransactionID	String	19	Transaction identifier
TransactionAmount	String	10	Dollar amount of transaction (Paymentech <= 99999.99). Format D.CC
OriginalAuthorizedAmount	String	10	Original dollar amount authorized. Format D.CC
SalesTaxAmount	String	12	Sales tax amount. Format D.CC
TipAmount	String	12	Tip amount. Format D.CC
CashBackAmount	String	12	Cash back amount. Format D.CC
ApprovalNumber	String	30	Issuer-assigned approval number. May not be returned for PIN Debit transactions or Credit Refund transactions.
ClerkNumber	String	10	Clerk number
ShiftID	String	10	Shift ID
ReferenceNumber	String	50	User-defined reference number. Numeric recommended. . For Check/ACH transactions, ReferenceNumber max length is 40 characters.
ReversalType	Enumeration	2	Type of reversal
MarketCode	Enumeration	2	Type of industry
AcquirerData	String	1000	Host-specific authorization response data. No action needed.
BillPaymentFlag	BooleanType	1	Specifies if this is a bill payment transaction

Name	Type	Max	Description
DuplicateCheckDisableFlag	BooleanType	1	Disables duplicate checking for this transaction. Recommended instead of DuplicateOverrideFlag.
DuplicateOverrideFlag	BooleanType	1	Disables duplicate checking for this transaction. It is recommended that DuplicateCheckDisableFlag be used.
DuplicateCheckExtended	BooleanType	1	If set to true, and if ReferenceNumber included, then ReferenceNumber will be included in the duplicate lookup for entire open batch (instead of card/amount only in the past 25 transactions)
RecurringFlag	BooleanType	1	Specifies if part of a recurring series of transactions
TicketNumber	String	50	User-defined ticket number. Numeric recommended.
CommercialCardCustomerCode	String	25	User-defined commercial card customer code
TransactionStatusCode	Enumeration	2	Status of transaction
TransactionStatus	String	25	Description of transaction status
TransactionSetupID	String	50	Unique GUID that identifies the TransactionSetupID. The Express platform generates this.
ApprovedAmount	String	10	Approved amount
ConvenienceFeeAmount	String	10	Convenience fee amount. Format D.CC. Vantiv only. NOTE: The max amount for this field with Vantiv is $99,999.99.
PartialApprovedFlag	BooleanType	1	Specifies whether software supports partial approvals
MerchantVerificationValue	String	10	MVV (TSYS, FDC, or Global only. Can be included or left out and picket up on authorization network profile.
CommercialCardResponseCode	String	2	Commercial card indicator (Vantiv, FDC, TSYS)
BalanceAmount	String	13	Balance amount. If negative, a minus sign may precede the amount. Format D.CC
BalanceCurrencyCode	String	3	Balance amount currency code 840 = US Dollars 124 = Canadian Dollars 978 = Euro
BillPayerAccountNumber	String	25	Bill payer account number
DCCRequested	Enumeration	2	On second pass request, indicates if the consumer opted out or in when prompted to pay in a foreign currency. Note that a value of Suspended means to disable DCC for that transaction.
ConversionRate	String	8	Conversion rate for the ForeignCurrencyCode. This field is sent in the response of the first pass transaction and must be included in the second pass transaction. The left-most digit represents the number of decimal positions that the decimal should be moved from the right of the conversion rate. Positions 2 through 8 are the conversion rate that was used (e.g. 61012345 is a conversion rate of 1.012345)
ForeignCurrencyCode	String	3	Currency code of the foreign currency. This field is send in the response of the first pass transaction and must be included in the second pass transaction.
ForeignTransactionAmount	String	12	Transaction amount shown in the foreign currency. This field is sent in the response of the first pass transaction and must be included in the second pass transaction. If the transaction amount is adjusted, the merchant must recalculate the ForeignTransactionAmount and include it in the second pass transaction.
SurchargeAmount	String	8	Surcharge fee amount (XML only; Vantiv only)
AlternateMCC	String	4	Alternate Merchant Category Code than what is configured on the Merchant's profile. Used for Buy Here Pay Here clients.
MerchantSuppliedTransactionID	String	36	An optional alphanumeric ID supplied by the Merchant. This should be a unique value, but this is up to the integrator, Express does not ensure its uniqueness. If this ID is supplied in a System Reversal, it must match the original Sale/Authorization and will be used lookup the original Transaction.
DuplicateTransactionID	String	19	May be returned with duplicate decline (XML only)
DuplicateApprovalNumber	String	30	May be returned with duplicate decline (XML only)
DuplicateHostItemID	String	25	May be returned with duplicate decline (XML only)
PINlessPOSConversionIndicator	String	1	PINless Debit conversion indicator (XML only; Vantiv only) Request Message 0 = Do not request conversion 1 = Request conversion

Name	Type	Max	Description
			Response Message: Y = Converted N = Not Converted
NetworkLabel	String	30	Network label where authorization was routed ("Visa", "MasterCard", etc.). May be returned based on processor.
NOCDate	String		Notification of Change Date. Returned on CheckQuery transactions.
DateProcessed	String		Date check was processed. Returned on CheckQuery transactions.
FundedDate	String		Date check funds were settled. Returned on CheckQuery transactions.
ReturnDate	String		Date check was returned. Returned on CheckQuery transactions.
MerchantDescriptor	String	25	Used by merchant to customize merchant name on cardholder statement (XML only; Vantiv only). Where applicable, returned in response if supplied on request.
MerchantDescriptorCity	String	13	Used by merchant to customize merchant city on cardholder statement (XML only; Vantiv only). Where applicable, returned in response if supplied on request.
MerchantDescriptorState	String	2	Used by merchant to customize merchant state on cardholder statement (XML only; Vantiv only). Where applicable, returned in response if supplied on request.
PaymentType	String	2	Payment type used for credential on file transactions
SubmissionType	String	2	Submission type used for credential on file transactions
NetworkTransactionID	String	40	Transaction identifier used for credential on file transactions
TypeOfGoods	String	1	Optional Ecommerce Type of Goods indicator. XML only (Vantiv). May be returned in the response if supplied. 1 = Physical 2 = Digital
PAR	String	35	Alphanumeric Payment Account Reference (PAR) token. If available, may be returned from the card issuer. XML only (Vantiv, TSYS, FDC).
SystemTraceAuditNumber	String	6	System Trace Audit Number that may be returned by the processor. XML only (Vantiv, TSYS).
RetrievalReferenceNumber	String	12	Retrieval Reference Number that may be returned by the processor. XML only (Vantiv).
SECCode	String	3	Standard Entry Class code, such as "ARC", "BOC", "RCK", "CCD", "PPD", "POP", "TEL", or "WEB".
CustomField[1-10]	String	128	Optional ACH custom input fields for ACH reporting purposes at Check Commerce (CustomField1 through CustomField10)
ACHNotes	String	1024	Pipe-delimited ACH notes. May be returned on ACH transactions depending on Check Commerce configuration.
ACHScores	String	1024	Pipe-delimited ACH scores. May be returned on ACH transactions depending on Check Commerce configuration.
DeferredAuthIndicator	BooleanType	1	Used to identify Store & Forward transactions (CreditCardSale, CreditCardAuthorization, CreditCardCredit, CreditCardReturn)
FraudSightRiskStatus	String	10	FraudSight Risk Status Code. Response only. Conditionally returned based on FraudSight merchant profile configuration. For Express/Vantiv clients, a single-digit numeric value is returned: 0=Pass, 1=Review, 2=Fail, 3=Unknown, 4=Fail (but do not deny – informational) For Express Gateway clients, a text value is returned: "Pass", "Review", "Fail", "Unknown"
FraudSightScore	String	6	FraudSight Score. Response only. Conditionally returned based on FraudSight merchant profile configuration.
WebSessionID	String	15	Customer Web Session Browser ID. Optional Ecommerce request value if using FraudSight (though not required for FraudSight).
OriginalTransactionID	String	19	Original TransactionID. May be returned in certain transaction scenarios.
OriginalTransactionType	String	60	Original TransactionType. May be returned in certain transaction scenarios.
DebtRepaymentFlag	BooleanType	1	Debt Repayment Flag. Optional request value only for use with Debt Repayment transactions. Supported for Vantiv, First Data-

Name	Type	Max	Description
			Nashville, and TSYS platforms only. Contact Developer Integrations for more information.
ReceiptEmail	String	80	Beta Agreement currently required to use this feature[1]. Contact merchant sales rep for additional information. Single cardholder email address where transaction receipt (for approval or decline) will be emailed. Only one receipt delivery option should be used. If both are present, then ReceiptEmail will be used by default.
ReceiptPhone	String	10	Beta Agreement currently required to use this feature[1]. Contact merchant sales rep for additional information. Single cardholder mobile phone number where transaction receipt (for approval or decline) will be sent as a text message. Only one receipt delivery option should be used. If both are present, then ReceiptEmail will be used by default.
Secure3DSVersion	String	1	Version of 3-D Secure protocol used. Required by MasterCard (Security Level Indicator-SLI) when using 3DS. 0 = Default/Null 1 = 3-D Secure Version 1.0 (3DS 1.0) 2 = EMV 3-D Secure (3DS 2.0)
Secure3DSDirectoryServerTranID	String	36	3-D Secure Directory Server Transaction ID. Required by MasterCard (Security Level Indicator-SLI) when using 3DS.
RCode	String	3	ACH Return Reason Code
BOE	String	2	Back-Office Exception Code

ReceiptEmail / ReceiptPhone[1] User understands and agrees that User is solely responsible for designing payment application functionality compliant with and as required under applicable federal and state law to disclose, attain and record the merchant customer acknowledgment and consent for the electronic delivery of the transaction receipts to such customers of merchants via e-mail or SMS text messages, as applicable which such requirements may include obtaining and recording the customers prior express consent for such deliveries and disclosure that message and data rates of the customers' carriers may apply to the Product messaging.

TransactionSetup Class

Name	Type	Max	Description
TransactionSetupID	String	50	Unique GUID that identifies the TransactionSetupID. The Express platform generates this. Expires after one of the following thresholds has been exceeded: 10 minutes, 5 decline attempts in Hosted Payments, or 1 approval in Hosted Payments (whichever comes first).
TransactionSetupMethod	Enumeration	2	Transaction method
Device	Enumeration	2	Entry device used
Embedded	BooleanType	2	Specifies if card entry page will be embedded in an application
CVVRequired	BooleanType	2	Specifies if user must enter CVV2 / CVC2 / CID values
AutoReturn	BooleanType	2	Specifies if approval detail will be auto-redirected to ReturnURL
CompanyName	String	50	Name of merchant
LogoURL	String	150	HTTPS URL of merchant logo to be displayed when Embedded flag is set to false
Tagline	String	150	Text displayed next to logo
WelcomeMessage	String	250	Text to display to cardholder (if not provided, a generic message will be displayed)
ReturnURL	String	150	Return URL link displayed after transaction completion or cancellation
ReturnURLTitle	String	100	Text to display for ReturnURL (if not provided, a generic "return to merchant" message will be displayed)
OrderDetails	String	250	Text to describe purchase
ProcessTransactionTitle	String	100	Text used for submit button label
ValidationCode	String	50	Unique code return from a TransactionSetup request and as part of the query string redirect during a successful transaction
DeviceInputCode	Enumeration	2	Device input code
CustomCss	String	10000	Custom CSS to style the payment page. See Custom CSS appendix.

Name	Type	Max	Description
HPType	BooleanType	2	Should only be used for Hosted Payments Premium integrations. Contact Developer Integrations for more information.

Enumerations

AccountType

Specifies the account type for an ACH TokenCreate request.

Name	Value	Description
Checking	0	Checking account
Savings	1	Savings account
CorporateChecking	3	Corporate checking account
CorporateSavings	4	Corporate savings account

AutoRentalAdjustmentCode

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

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

AutoRentalDistanceUnit

AutoRentalDistanceUnit specifies the distance unit of measurement.

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

AutoRentalVehicleClassCode

AutoRentalVehicleClassCode specifies the classification of rental vehicle.

Name	Value	Description
Unused	0	Unused
Mini	1	Mini
Subcompact	2	Subcompact
Economy	3	Economy
Compact	4	Compact
Midsize	5	Midsize
Intermediate	6	Intermediate
Standard	7	Standard
Fullsize	8	Fullsize
Luxury	9	Luxury
Premium	10	Premium
MiniVan	11	MiniVan
Van12Passenger	12	12-Passenger Van
MovingVan	13	Moving Van
Van15Passenger	14	15-Passenger Van

Name	Value	Description
CargoVan	15	Cargo Van
Truck12Foot	16	12-Foot Truck
Truck20Foot	17	20-Foot Truck
Truck24Foot	18	24-Foot Truck
Truck25Foot	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
TruckLarge	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

BatchCloseType

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

Name	Value	Description
Regular	0	Regular
Force	1	Force

BatchGroupingCode

BatchGroupingCode specifies which Batch you are querying against.

Name	Value	Description
FullBatch	0	Full batch
SingleBatch	1	Single batch

BatchIndexCode

BatchIndexCode specifies which Batch you are querying against.

Name	Value	Description
Current	0	Current open batch
FirstPrevious	1	First previous batch (NOTE: this option is no longer valid)

BatchQueryType

This enumeration is reserved for future use.

Name	Value	Description
Total	0	Total
Item	1	Item

BooleanType

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

Name	Value	Description
False	0	False
True	1	True

CardholderPresentCode

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

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 by which the Card Number or Track Data was acquired.

Name	Value	Description
Default	0	Default
Unknown	1	Unknown
MagstripeRead	2	Magstripe Read
ContactlessMagstripeRead	3	Contactless Magstripe Read
ManualKeyed	4	Manual Keyed
ManualKeyedMagstripeFailure	5	Manual Keyed Magstripe Failure
ChipRead	6	Chip Read
ContactlessChipRead	7	Contactless Chip Read
ManualKeyedChipReadFailure	8	Manual Keyed Chip Read Failure
MagstripeReadChipReadFailure	9	Magstripe Read Chip Read Failure
MagstripeReadNonTechnicalFallback	10	Magstripe Read Non-Technical Fallback

CardLogo

CardLogo specifies the Card Logo/Card Brand as a numeric value.

Name	Value	Description
Visa	1	Visa
MasterCard	2	MasterCard
Amex	3	Amex
Discover	4	Discover
Diners	5	Diners Club
StoredValue	6	StoreCard
Other	7	Other
JCB	8	JCB
CarteBlanche	9	Carte Blanche
Interac	10	Interac
UnionPay	11	Union Pay

CardPresentCode

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

Name	Value	Description
Default	0	Default
Unknown	1	Unknown
Present	2	Present
NotPresent	3	Not Present

CheckType

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

Name	Value	Description
Personal	0	Personal
Business	1	Business

ConsentCode

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

Name	Value	Description
NotSpecified	0	NotSpecified
FaceToFace	1	Face to face
Phone	2	Phone
Internet	3	Internet

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.

Name	Value	Description
UseDefault	0	Use Default
Not Provided	1	Not Provided
Provided	2	Provided
Illegible	3	Illegible
CustomerIllegible	4	Customer Illegible

CVVResponseType

CVVResponseType specifies the type of CVV Response codes that are returned by the host.

- The Extended value only applies to Visa cards.

Name	Value	Description
Regular	0	Regular
Extended	1	Extended

DCCRequested

Specifies if the consumer opted in or out when prompted to pay in their native currency.

- Suspend indicates to disable DCC for the current transaction.

Name	Value	Description
OptOut	0	Opt out and pay in local currency
OptIn	1	Opt in and pay in foreign currency
Suspended	2	Disable DCC for this transaction

DDAAccountType

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

Name	Value	Description
Checking	0	Checking
Savings	1	Savings

Device

Device specifies the entry device used for the transaction.

Name	Value	Description
Default	0	Default
MagtekEncryptedSwipe	1	Magtek Encrypted Swipe
EncryptedInputDevice	2	Encrypted Input Device (Reserved for future use)

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

ElectronicCommerceIndicator

The ElectronicCommerceIndicator is used to identify eCommerce transaction values

Name	Value	Description
Default	0	Default
NotUsed	1	Not used
Single	2	Single
Recurring	3	Recurring
Installment	4	Installment
SecureElectronicCommerce	5	Secure Electronic Commerce
NonAuthenticatedSecureTransaction	6	Non-Authenticated Secure Transaction
NonAuthenticatedSecureECommerceTransaction	7	Non-Authenticated Secure eCommerce Transaction. Use for all Express eCommerce transactions.
NonSecureECommerceTransaction	8	Non-Secure eCommerce Transaction
AmericanExpressToken	9	American Express Token (In-App Purchase)

EncryptedFormat

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

Name	Value	Description
Default	0	Default

Name	Value	Description
Format1	1	Magtek
Format2	2	Ingenico DPP (device retired)
Format3	3	Ingenico On-Guard (Reserved for future use)
Format4	4	ID Tech
Format5	5	Verifone Vx devices using MercuryPay format
Format6	6	Infinite Peripherals
Format7	7	Verifone Mx devices
Format8	8	Ziosk Magtek Intellihead
Format9	9	Verifone Vx devised using CXPI 12.X
Format10	10	Ingenico Tetra (XML only)
Format11	11	Ingenico Generic TDES (XML only)

ExtendedBooleanType

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

Name	Value	Description
Null	-1	Null
False	0	False
True	1	True

ExtendedRunFrequency

ExtendedRunFrequency specifies the how often an event should occur.

Name	Value	Description
Null	-1	Null
OneTimeFuture	0	One Time Future
Daily	1	Daily
Weekly	2	Weekly
BiWeekly	3	BiWeekly
Monthly	4	Monthly
BiMonthly	5	BiMonthly
Quarterly	6	Quarterly
SemiAnnually	7	SemiAnnually
Yearly	8	Yearly

ExtendedStatusType

ExtendedStatusType specifies current status of an item.

Name	Value	Description
Null	-1	Null
Active	0	Active
Disabled	1	Disabled
Removed	2	Removed

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

Healthcare Account Type specifies the type of healthcare account.

Name	Value	Description
NotSpecified	0	Not specified
Savings	1	Savings

Name	Value	Description
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]AmountSign

Healthcare Amount Sign specifies the healthcare amount sign.

Name	Value	Description
Positive	0	Positive
Negative	1	Negative

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
Healthcare	2	Healthcare (this Healthcare amount type is required if including Healthcare/IIAS amount details)
Transit	3	Transit
Copayment	4	Copayment
OriginalAmount	5	Original Amount
PartialAuthorizedAmount	6	Partial Authorized Amount
Prescription	7	Prescription
Vision	8	Vision
Clinic	9	Clinic
Dental	10	Dental
CashOver	11	Cash Over
OriginalCashOver	12	Original Cash Over

LodgingChargeType

LodgingChargeType specifies a hotel charge type.

Name	Value	Description
Default	0	Default
Restaurant	1	Restaurant
GiftShop	2	GiftShop

LodgingPrestigiousPropertyCode

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

Name	Value	Description
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

LodgingSpecialProgramCode

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

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

MarketCode

MarketCode specifies the industry type of the merchant.

- Set this to a value of Default to use the MarketCode from the merchant profile.

Name	Value	Description
Default	0	Default
AutoRental	1	Auto Rental
DirectMarketing	2	Direct Marketing (MOTO)
ECommerce	3	ECommerce
FoodRestaurant	4	Food / Restaurant
HotelLodging	5	Hotel / Lodging
Petroleum	6	Petroleum (NOTE: not currently used)
Retail	7	Retail
QSR	8	QSR (NOTE: not currently used)
Grocery	9	Grocery (NOTE: not currently used)

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
Default	0	Default
NotUsed	1	Not used
Single	2	Single
Recurring	3	Recurring
Installment	4	Installment
SecureElectronicCommerce	5	Secure Electronic Commerce (3-D Secure 2.0, Vantiv only)
NonAuthenticatedSecureTransaction	6	Non-Authenticated Secure Transaction (3-D Secure 2.0 attempted, but authentication data not provided, Vantiv only)
NonAuthenticatedSecureECommerceTransaction	7	Non-Authenticated Secure eCommerce Transaction. Use for all Express eCommerce transactions.
NonSecureECommerceTransaction	8	Non-Secure eCommerce Transaction
AmericanExpressToken	9	American Express Token (In-App Purchase)

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.

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
AutoUpdateEnabled	1	Auto updated enabled
AutoUpdateDisabled	2	Auto updated disabled

PASSUpdaterStatus

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
UpdateInProgress	1	Updated 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
InvalidInfo	9	Invalid Info
NoResponse	10	No response
NotAllowed	11	Not allowed
Error	12	Error
PASSUpdaterDisabled	13	PASS Updater disabled
NotUpdated	14	Not updated
Issuer Corrected	15	Issuer corrected a previous update
Fraud	16	Fraud has been reported at Merchant (Amex only)
Inactivity	17	Inactivity (No spend on Card at Merchant over a given period) (Amex)
Merchant Blocked	18	Merchant / Seller blocked by Cardmember (Amex only)
Inactive Seller	19	Inactive Seller (Merchant is not active and has been cancelled) (Amex only)

PaymentAccountType

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

Name	Value	Description
CreditCard	0	Credit Card
Checking	1	Checking
Savings	2	Savings
ACH	3	ACH
Other	4	Other

PaymentType

PaymentType indicates the type of credentials-on-file/card on file payment being processed.

Name	Value	Description
NotUsed	0	Not used
Recurring	1	Recurring
Installment	2	Installment
CardHolderInitiated	3	Cardholder Initiated
CredentialOnFile	4	Credential on File

ReversalType

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

Name	Value	Description
System	0	System
Full	1	Full
Partial	2	Partial

RunFrequency

RunFrequency specifies the how often an event should occur.

Name	Value	Description
OneTimeFuture	0	One-time future
Daily	1	Daily
Weekly	2	Weekly
BiWeekly	3	Bi-weekly
Monthly	4	Monthly
BiMonthly	5	Bi-monthly
Quarterly	6	Quarterly
SemiAnnually	7	Semi-annually
Yearly	8	Yearly
MonthlyLastDay	9	Last day of the month. After the initial RunStartDate transaction, all subsequent transactions will automatically process on the last day of each month. This option is supported in the XML interface only.

ScheduledTaskRunStatus

ScheduledTaskRunStatus specifies the current run status of a task.

Name	Value	Description
Null	-1	Null
Pending	0	Pending
Running	1	Running
Completed	2	Completed
Error	3	Error
SystemRetry	4	System Retry
ManualRetry	5	Manual Retry

StatusType

StatusType specifies current status of an item.

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

SubmissionType

SubmissionType indicates the type of credentials-on-file/card on file submission being sent.

Name	Value	Description
NotUsed	0	Not used
Initial	1	Initial
Subsequent	2	Subsequent
Resubmission	3	Resubmission
ReAuthorization	4	Re-Authorization
DelayedCharges	5	Delayed Charges

Name	Value	Description
NoShow	6	No-Show

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
MagstripeReader	3	Magstripe Reader
ContactlessMagstripeReader	4	Contactless Magstripe Reader
KeyEntered	5	Key entered
ChipReader	6	Chip Reader
ContactlessChipReader	7	Contactless Chip Reader

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.

Name	Value	Description
Default	0	Default
Format1	1	Format 1
Format2	2	Format 2
Format3	3	Format 3
Format4	4	Format 4
Format5	5	Format 5

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.

Name	Value	Description
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.

Name	Value	Description
Unknown	0	Unknown
PointOfSale	1	Point of Sale
ECommerce	2	ECommerce
MOTO	3	MOTO
FuelPump	4	Fuel Pump (Reserved for future use)
ATM	5	ATM
Voice	6	Voice

Name	Value	Description
Mobile	7	Mobile
WebSiteGiftCard	8	Web Site Gift Card. Used for Vantiv Gift Card Program Virtual Gift functionality only. XML only.

TokenProvider

TokenProvider specifies the provider of the tokenization solutions.

Name	Value	Description
Null	0	Null
ExpressPASS	1	Express PASS
OmniToken	2	OmniToken
Paymetric	3	Paymetric
TransArmor	4	TransArmor

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 (please use 9 / Sale instead)
CheckSale	6	CheckSale (Reserved for future use)
PaymentAccountCreate	7	PaymentAccountCreate
PaymentAccountUpdate	8	PaymentAccountUpdate
Sale	9	Sale that supports both Credit and PIN Debit

TransactionStatusCode

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

Name	Value	Description
InProcess	0	In Process
Approved	1	Approved
Declined	2	Declined
Duplicate	3	Duplicate
Voided	4	Voided
Authorized	5	Authorized
AuthCompleted	6	Authorization Completed
Reversed	7	Reversed
Success	8	Success
Returned	9	Returned (ACH only)
Pending	10	Pending (ACH only)
Queued	11	Queued
Unknown	12	Unknown
Error	13	Error
Originated	14	Originated (ACH only)
Settled	15	Settled
PartialApproved	16	Partial Approved
Rejected	17	Rejected

Name	Value	Description
DCCRequested	20	Dynamic Currency Conversion requested. XML only.

WalletType

WalletType specifies the mobile wallet used for an In App transaction.

Name	Value	Description
Null	0	Null
Android	1	Android Wallet
Apple	2	Apple Wallet
Samsung	3	Samsung Wallet

Appendix 1: Response Codes

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.

Code	Description
0	Approved / Success
5	Partial Approval
7	DCC Requested
8	Non-Financial Card
20	Declined
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 Authorized
120	Out of Balance
1001	Communication Error
1002	Host Error
1009	Error

AVS Response Codes

Address Verification Service response codes from card issuers outside the U.S. (for cardholder billing addresses outside the U.S.) may provide inconsistent results.

Code	Description	Domestic	International
A	Street Address matches, Postal Code does not match.	X	X
B	Street Address matches, Postal Code not verified (due to incompatible format).	X	X
C	Street Address and Postal Code not verified (due to incompatible format).	X	X
D	Street Address and Postal Code match.		X
E	Edit error or ineligible transaction for AVS.	X	X
F	Street Address and Postal Code match (UK only)		X
G	Address information not verified.		X

Code	Description	Domestic	International
I	Address information not verified.		X
M	Street Address and Postal Code match.		X
N	Street Address does not match, Postal Code does not match.	X	X
P	Postal Code matches, Street Address not verified (due to incompatible format).	X	X
R	System unavailable or timeout.	X	X
S	AVS currently not supported.	X	
T	Postal Code (generally 9-digit) matches, Street Address does not match.	X	
U	Address information unavailable.	X	
W	Postal Code (generally 9-digit) matches, Street Address does not match.	X	
X	Street Address and Postal Code (generally 9-digit) match.	X	
Y	Street Address and Postal Code (generally 5-digit) match.	X	X
Z	Postal Code (generally 5-digit) matches, Street Address does not match.	X	X
0	Address verification not requested (or requested but transaction was declined (TSYS)).	X	X

NOTE: Effective April 14, 2023, Visa may no longer return the following AVS values: B, C, G, I, S D, F M, X, P, W

CVV Response Codes

Code	Description
M	Match
N	No Match
P	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)
A	CAVV passed validation – attempt (US issued card, non-US merchant, issuer authentication server not available)
B	CAVV passed validation – information only, no liability shift

Commercial Card Response Codes

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

Code	Description
B	Business Card
D	Visa Commerce (Reserved for future use)
E	Visa B2B settlement match edits eligible card
L	Visa B2B settlement match edits eligible card
R	Corporate Card
S	Purchasing Card
T	Unable to obtain information
0	Non-Commercial Card
Space	Non-Commercial Card or request error

Appendix 2: Sample Code

HealthCheck

```
[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>"
sXml = sXml & "<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
```

Appendix 3: Test Trigger Amounts and Values

Please visit the following links for the latest triggers for various test processors and transaction types within the Express Cert environment. Triggers to return various ExpressResponseCode, AVSResponseCode, and CVVResponseCode values are included on those pages.

https://developerengine.fisglobal.com/apis/express/features/test-triggers

https://developerengine.fisglobal.com/apis/express/express-xml/null

Appendix 4: Other Details

Terminal Settings

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

Interface URLs

For a list of Express Platform XSD and WSDL links please reviewhttps://developerengine.fisglobal.com/apis/express/express-xml/xsd

Interface	Platform	Type	URL
Transaction	Cert	XML	https://certtransaction.elementexpress.com/
Transaction	Cert	SOAP (no longer being updated)	https://certtransaction.elementexpress.com/express.asmx
Reporting	Cert	XML	https://certreporting.elementexpress.com/
Reporting	Cert	SOAP (no longer being updated)	https://certreporting.elementexpress.com/express.asmx
Services	Cert	XML	https://certservices.elementexpress.com/
Services	Cert	SOAP (no longer being updated)	https://certservices.elementexpress.com/express.asmx

Interface	Supported Request Methods
Transaction	CreditCard*, Batch*, Check*, DebitCard*, EBT*, EnhancedBINQuery, GiftCard*, HealthCheck, TransactionSetup*,
Reporting	TransactionQuery
Services	PaymentAccount*, ScheduledTask*, TokenCreate*

Hosted Payments Custom CSS

To customize the styling of the Hosted Payment page, you may pass in custom CSS to the TransactionSetup method in the CustomCss field.

When the page loads, it overrides the default styling with the custom CSS.

For security reasons, the CSS must be scrubbed and validated against a whitelist of allowed CSS attributes and values as well as class names and element names. See Custom CSS Whitelist below.

Use the browser developer tool (press F12) to inspect the rendered payment page to find html element names, class names, and css attributes you want to override.

Here is a simple example of custom CSS that goes a long way to change the background color of the page, the color of the font, the font displayed in dropdowns and textboxes and the color of the font on the payment button. This alone will get you close to having a more seamless user experience.

Note: The important declaration is supported when you want to force a css rule to apply.

Custom CSS Example:

```
body {
  font-family: Arial; font-size: 12px; color: #000099;
  background-color: #e6e6ff;
}
.selectOption {
  font-family:Arial;
```

```
  }
  .inputText {
    font-family:Arial;
  }
  .buttonEmbedded:link {
    color: blue;
  }
```

Custom CSS Whitelist

For security reasons, the custom CSS must be scrubbed and validated against a whitelist of allowed CSS attributes and values as well as class names and element names.

Use the browser developer tool (press F12) to inspect the rendered payment page to find html elements, class names, and css attributes you want to override. Most of those various items are allowed in the Vantiv whitelist.

If you send in custom css and your css rules are not taking effect, use the developer tool on the browser to inspect the element in question and see what css rules were actually applied. If your css rule is not being applied, check to make sure it's supported in the whitelist and that the css syntax is correct including case sensitivity.

Contact the Integrated Payments Developer Integrations team at certification2@fisglobal.com to obtain the CSS whitelist information.

Developer Integrations Team

To contact the Express Developer Integrations team, please email certification2@fisglobal.com.