eSelect PLUS

Merchant Integration Guide

MasterCard Wallet
MasterPass Transaction DTD
v.1.0.0

Moneris SOLUTIONS

Revision Number	**Date**	**Description**
V 1.0.0	August 28, 2013	-Initial Draft

Table of Contents

****** PLEASE READ CAREFULLY******

You have a responsibility to protect cardholder and merchant related confidential account information. Under no circumstances should ANY confidential information be sent via email while attempting to diagnose integration or production issues. When sending sample files or code for analysis by Moneris staff, all references to valid card numbers, merchant accounts and transaction tokens should be removed and or obscured. Under no circumstances should live cardholder accounts be used in the test environment.

1. About this Documentation

The eSELECTplus payment gateway supports credit card transactions in XML format over the HTTPS protocol. This document describes the detailed information for using the MasterPass to process transactions. In particular, it describes the XML format for sending transactions and the corresponding responses you will receive. If you are also interested in being able to collect payment from credit card without MasterPass, please contact us for XML documentation at eproducts@moneris.com.

2. System and Skill Requirements

In order to use .NET your system will need to have the following:

1. A web server with an SSL certificate
2. Server service capable of server to server post
3. Port 443 open for bi-directional communication

As well, you will need to have the following knowledge and/or skill set:

1. Knowledge of XML creation and parsing

3. What is the Process I will need to follow?

You will need to follow these steps.

1. Do the required development as outlined in this document
2. Test your solution in the test environment
3. Activate your store
4. Make the necessary changes to move your solution from the test environment into production as outlined in this document

Note:

It is important to note that all Merchants and Service Providers that store, process, or transmit cardholder data must comply with PCI DSS and the Card Association Compliance Programs. However, certification requirements vary by business and are contingent upon your "Merchant Level" or "Service Provider Level". Failure to comply with PCI DSS and the Card Association Compliance Programs may result in a Merchant being subject to fines, fees or assessments and/or termination of processing services. Non-compliant solutions may prevent merchants boarding with Moneris Solutions.

As a Moneris Solutions client or partner using this method of integration, your solution must demonstrate compliance to the Payment Card Industry Data Security Standard (PCI DSS) and/or the Payment Application Data Security Standard (PA DSS). These standards are designed to help the cardholders and merchants in such ways as they ensure credit card numbers are encrypted when transmitted/stored in a database and that merchants have strong access control measures.

For further information on PCI DSS and PA DSS requirements, please visit http://www.pcisecuritystandards.org.

For more information on how to get your application PCI-DSS compliant, please contact our Integration Specialists and visit https://developer.moneris.com to download the PCI-DSS Implementation Guide.

4. Transaction Types and Transaction Flow

Below is a list of transactions supported by the MasterPass XML, other terms used for the transaction type are indicated in brackets.

paypass_send_shopping cart: Mandatory call to Moneris to obtain *MPRequestToken* and *MPRedirectUrl*. Your customers must be redirect to Url specified in *MPRedirectUrl* to proceed with checkout.

paypass_retrieve_checkout_data: Mandatory call to Moneris after customer is redirect back to your site. This call allows you to obtain customer profile details such as billing address, shipping address, masked card number, expiry date, customer contact information and cavv value.

paypass_purchase: Call to Moneris to obtain funds from the MasterPass *oauthtoken* and ready them for deposit into the merchant account. This call can only made after making *paypass_retreive_checkout_data* call.

paypass_preauth: Call to Moneris to verify funds on the MasterPass *oauthtoken* and reserve those funds for your merchant account. The funds are locked for a specified amount of time, based on the card issuer. This call can only made after making *paypass_retreive_checkout_data* call. To retrieve the funds from this call so that they may be settled in the merchant's account, a completion must be performed.

paypass_completion: Call to Moneris to obtain funds reserved by *paypass_preauth or paypass_cavv_preauth*. This transaction call retrieves the locked funds and readies them for settlement into the merchant's account. This call must be made typically within 72 hours of performing *paypass_preauth* or *paypass_cavv_preauth*.

paypass_purchasecorrection: Call to Moneris to void the *paypass_purchase or paypass_completion* the same day* that they occurred on.

paypass_refund: Call to Moneris to refund against a *paypass_purchase* or *paypass_completion* to refund any part or all of the transaction.

paypass_cavv_purchase: Verified by Visa or MasterCard SecureCode transaction call to Moneris using *Cavv* value to obtain funds from the MasterPass *oauthtoken* and ready them for deposit into the merchant account. This call can only made after making *paypass_retreive_checkout_data* call. Cavv value can be obtained from *paypass_retrieve_checkout_data* or by performing *paypass_txn* call.

paypass_cavv_preauth: Verified by Visa/MasterCard SecureCode transaction call to Moneris using *Cavv* value to verify funds on the MasterPass *oauthtoken* and reserve those funds for your merchant account. The funds are locked for a specified amount of time, based on the card issuer. This call can only made after making *paypass_retreive_checkout_data* call. To retrieve the funds from this call so that they may be settled in the merchant's account, a completion must be performed.

paypass_txn: Optional call to Moneris to perform Verified by Visa or MasterCard SecureCode MPI transaction to obtain *Cavv* value. This call should be performed if you don't receive *AuthenticationOptionsCAvv* value in response after performing *paypass_retrieve_checkout_data.*

NOTE

* A Void can be performed against a transaction as long as the batch that contains the original transaction remains open. When using the automated closing feature Batch Close occurs daily between 10 – 11 pm EST.

Transaction Flow for MasterPass Transaction

1. Once your customer has selected MasterPass and proceeds to pay, you must make paypass_send_shopping_cart call to Moneris to obtain MPRequestToken and MPRedirectUrl.
2. Your website will then redirect to your customer to url specified in MPRedirectUrl from step 1.
3. Once customer completes their process on MasterPass, MasterPass will redirect customer back to your site along with response.
4. Using variables from step 3 response, you must then make paypass_retrieve_checkout_data call to Moneris. This call will verify the response token from MasterPass response and it will provide you with customer profile details from MasterPass.
5. OPTIONAL: Calculate shipping cost using data from paypass_retrieve_checkout_data call and add it to the total amount
6. Now, make a paypass_purchase or paypass_preauth call to Moneris to charge the card and obtain funds.

NOTE If paypass_retrieve_checkout_data provides you with CAVV data, you can perform paypass_cavv_purchase or paypass_cavv_preauth transaction.

5. What Information do I need to include in a Transaction Request.

A transaction request contains various fields depending on the type of transaction that is being performed. The following table illustrates the fields that you can send for each transaction type. For a full description of each field please refer to Appendix A. Definition of Request Fields. Please note that the fields must be in the correct order.

Required Fields	
Transaction Type	**Fields**
	MasterPass Administrative Transactions
MasterPass Send Shopping cart	paypass_send_shopping_cart (subtotal,suppress_shipping_address)
MasterPass Retrieve Shopping cart	paypass_retrieve_checkout_data (oauth_token, oauth_verifier, checkout_resource_url)
	Basic Transaction Types
MasterPass Purchase	paypass_purchase (order_id, cust_id [optional], amount, mp_request_token, crypt_type, dynamic_descriptor [optional])
MasterPass Preauth	paypass_preauth (order_id, cust_id [optional], amount, mp_request_token, crypt_type, dynamic_descriptor [optional])
MasterPass Purchase Correction	paypass_purchasecorrection (order_id, txn_number, crypt_type)
MasterPass Refund	paypass_refund (order_id, amount, txn_number, crypt_type, dynamic_descriptor[optional])
MasterPass Completion	paypass_completion (order_id, comp_amount, txn_number, crypt_type, dynamic_descriptor[optional])
	Verifed by Visa (VbV)/MasterCard SecureCode (MCSC) Transactions
MasterPass MPI	paypass_txn (xid, amount, mp_request_token, MD[optional], merchantUrl, accept, userAgent, currency [optional], recurFreq[optional], recurEnd[optional], install [optional])
MasterPass VbV/MCSC Purchase	paypass_cavv_purchase (order_id, cavv, cust_id [optional], amount, mp_request_token, crypt_type, dynamic_descriptor [optional])
MasterPass VbV/MCSC Preauth	paypass_cavv_preauth (order_id, cavv, cust_id [optional], amount, mp_request_token, crypt_type, dynamic_descriptor [optional])

* Please refer to Appendix A. Definition of Request Fields for field definitions.

NOTE

The Transaction Types (ex. paypass_send_shopping_cart or paypass_purchase) is case sensitive and must be all in lower case.

The amount must have two decimal places and please do not include the $ sign. Also, if it is less than 1.00 it must contain a leading 0 (ex. 0.10). Minimum amount is 0.01 and maximum is 9999999.99.

6. Request DTD

```
<!-- The MasterPass Request DTD v1.0.1 -->
<!-- Main Elements -->
<!ELEMENT request (store_id, api_token, (paypass_send_shopping_cart |
paypass_retrieve_checkout_data | paypass_purchase | paypass_cavv_purchase |
paypass_preauth | paypass_cavv_preauth ))>

<!ELEMENT store_id (#PCDATA)>
<!ELEMENT api_token (#PCDATA)>

<!--The following are paypass transactions -->
<!ELEMENT paypass_send_shopping_cart (subtotal,suppress_shipping_address)>
<!ELEMENT paypass_retrieve_checkout_data (oauth_token, oauth_verifier,
checkout_resource_url)>
<!ELEMENT paypass_purchase (order_id, cust_id?, amount, mp_request_token, crypt_type,
avs_info?, dynamic_descriptor?)>
<!ELEMENT paypass_cavv_purchase (order_id, cavv, cust_id?, amount, mp_request_token,
crypt_type, avs_info?, dynamic_descriptor?)>
<!ELEMENT paypass_preauth (order_id, cust_id?, amount, mp_request_token, crypt_type,
avs_info?, dynamic_descriptor?)>
<!ELEMENT paypass_cavv_preauth (order_id, cavv, cust_id?, amount, mp_request_token,
crypt_type, avs_info?, dynamic_descriptor?)>
<!ELEMENT paypass_purchasecorrection (order_id, txn_number, crypt_type)>
<!ELEMENT paypass_refund (order_id, amount, txn_number, crypt_type,
dynamic_descriptor?)>
<!ELEMENT paypass_completion (order_id, comp_amount, txn_number, crypt_type,
dynamic_descriptor?)>
<!ELEMENT paypass_txn (xid, amount, mp_request_token, MD?, merchantUrl, accept,
userAgent)>

<!ELEMENT order_id (#PCDATA)>
<!ELEMENT cust_id (#PCDATA)>
<!ELEMENT cavv (#PCDATA)>
<!ELEMENT amount (#PCDATA)>
<!ELEMENT subtotal (#PCDATA)>
<!ELEMENT suppress_shipping_address (#PCDATA)>
<!ELEMENT oauth_token (#PCDATA)>
<!ELEMENT oauth_verifier (#PCDATA)>
<!ELEMENT checkout_resource_url (#PCDATA)>
<!ELEMENT mp_request_token (#PCDATA)>
<!ELEMENT crypt_type (#PCDATA)>
<!ELEMENT dynamic_descriptor (#PCDATA)>
<!ELEMENT comp_amount (#PCDATA)>
<!ELEMENT xid (#PCDATA)>
<!ELEMENT MD (#PCDATA)>
<!ELEMENT merchantUrl (#PCDATA)>
<!ELEMENT accept (#PCDATA)>
<!ELEMENT userAgent (#PCDATA)>

<!-- AVS Check -->
<!ELEMENT avs_info (avs_street_number, avs_street_name, avs_zipcode, avs_email?,
avs_hostname?, avs_browser?, avs_shiptocountry?, avs_shipmethod?, avs_merchprodsku?,
avs_custip?, avs_custphone?)>
<!ELEMENT avs_street_number (#PCDATA)>
<!ELEMENT avs_street_name (#PCDATA)>
<!ELEMENT avs_zipcode (#PCDATA)>
<!ELEMENT avs_email (#PCDATA)>
```

```
<!ELEMENT avs_hostname (#PCDATA)>
<!ELEMENT avs_browser (#PCDATA)>
<!ELEMENT avs_shiptocountry (#PCDATA)>
<!ELEMENT avs_shipmethod (#PCDATA)>
<!ELEMENT avs_merchprodsku (#PCDATA)>
<!ELEMENT avs_custip (#PCDATA)>
<!ELEMENT avs_custphone (#PCDATA)>
```

7. Response DTD

```
<!-- The MasterPass Response DTD v1.0.1 -->
<!-- Main Elements -->

<!ELEMENT response receipt+>
<!ELEMENT receipt (ResponseCode, Message, ReceiptId?, ReferenceNum?, ISO?, AuthCode?,
TransTime?, TransDate?, TransType?, Complete?, TransAmount?, CardType?, TransID?,
TimedOut?, Ticket?, MPRequestToken?, MPRedirectUrl?, PayPassInfo?)>

<!ELEMENT ReceiptId (#PCDATA)>
<!ELEMENT ReferenceNum (#PCDATA)>
<!ELEMENT ResponseCode (#PCDATA)>
<!ELEMENT ISO (#PCDATA)>
<!ELEMENT AuthCode (#PCDATA)>
<!ELEMENT TransTime (#PCDATA)>
<!ELEMENT TransDate (#PCDATA)>
<!ELEMENT TransType (#PCDATA)>
<!ELEMENT Complete (#PCDATA)>
<!ELEMENT Message (#PCDATA)>
<!ELEMENT TransAmount (#PCDATA)>
<!ELEMENT CardType (#PCDATA)>
<!ELEMENT TransID (#PCDATA)>
<!ELEMENT TimedOut (#PCDATA)>
<!ELEMENT Ticket (#PCDATA)>
<!ELEMENT MPRequestToken (#PCDATA)>
<!ELEMENT MPRedirectUrl (#PCDATA)>

<!-- The following are only applicable to MasterPass transactions -->
<!ELEMENT PayPassInfo (CardBrandId?, CardBrandName?, CardBillingAddressCity?,
CardBillingAddressCountry?, CardBillingAddressCountrySubdivision?,
CardBillingAddressLine1?, CardBillingAddressLine2?, CardBillingAddressPostalCode?,
CardBillingAddressRecipientPhoneNumber?, CardBillingAddressRecipientName?,
CardCardHolderName?, CardExpiryMonth, CardExpiryYear?, TransactionId?,
ContactEmailAddress?, ContactFirstName?, ContactLastName?, ContactPhoneNumber?,
ShippingAddressCity?, ShippingAddressCountry?, ShippingAddressCountrySubdivision?,
ShippingAddressLine1?, ShippingAddressLine2?, ShippingAddressPostalCode?,
ShippingAddressRecipientName?, ShippingAddressRecipientPhoneNumber?,
PayPassWalletIndicator?, AuthenticationOptionsAuthenticateMethod?,
AuthenticationOptionsCardEnrollmentMethod?, AuthenticationOptionsCAvv?,
AuthenticationOptionsEciFlag?, AuthenticationOptionsPaResStatus?,
AuthenticationOptionsSCEnrollmentStatus?,AuthenticationOptionsSignatureVerification?,
AuthenticationOptionsXid?, CardAccountNumber?)>

<!ELEMENT CardBrandId (#PCDATA)>
<!ELEMENT CardBrandName (#PCDATA)>
<!ELEMENT CardBillingAddressCity (#PCDATA)>
<!ELEMENT CardBillingAddressCountry (#PCDATA)>
<!ELEMENT CardBillingAddressCountrySubdivision (#PCDATA)>
```

```
<!ELEMENT CardBillingAddressLine1 (#PCDATA)>
<!ELEMENT CardBillingAddressLine2 (#PCDATA)>
<!ELEMENT CardBillingAddressPostalCode (#PCDATA)>
<!ELEMENT CardBillingAddressRecipientPhoneNumber (#PCDATA)>
<!ELEMENT CardBillingAddressRecipientName (#PCDATA)>
<!ELEMENT CardCardHolderName (#PCDATA)>
<!ELEMENT CardExpiryMonth (#PCDATA)>
<!ELEMENT CardExpiryYear (#PCDATA)>
<!ELEMENT ContactEmailAddress (#PCDATA)>
<!ELEMENT ContactFirstName (#PCDATA)>
<!ELEMENT ContactLastName (#PCDATA)>
<!ELEMENT ContactPhoneNumber (#PCDATA)>
<!ELEMENT ShippingAddressCity (#PCDATA)>
<!ELEMENT ShippingAddressCountry (#PCDATA)>
<!ELEMENT ShippingAddressCountrySubdivision (#PCDATA)>
<!ELEMENT ShippingAddressLine1 (#PCDATA)>
<!ELEMENT ShippingAddressLine2 (#PCDATA)>
<!ELEMENT ShippingAddressPostalCode (#PCDATA)>
<!ELEMENT ShippingAddressRecipientName (#PCDATA)>
<!ELEMENT ShippingAddressRecipientPhoneNumber (#PCDATA)>
<!ELEMENT PayPassWalletIndicator (#PCDATA)>
<!ELEMENT AuthenticationOptionsAuthenticateMethod (#PCDATA)>
<!ELEMENT AuthenticationOptionsCardEnrollmentMethod (#PCDATA)>
<!ELEMENT CardAccountNumber (#PCDATA)>
<!ELEMENT AuthenticationOptionsEciFlag (#PCDATA)>
<!ELEMENT AuthenticationOptionsPaResStatus (#PCDATA)>
<!ELEMENT AuthenticationOptionsSCEnrollmentStatus (#PCDATA)>
<!ELEMENT AuthenticationOptionsSignatureVerification (#PCDATA)>
<!ELEMENT AuthenticationOptionsXid (#PCDATA)>
<!ELEMENT AuthenticationOptionsCAvv (#PCDATA)>
<!ELEMENT TransactionId (#PCDATA)>


<!-- The following are only applicable to MasterPass Txn request -->
<!ELEMENT MpiResponse (type?, success?, message?,PaReq?, TermUrl?, MD?, ACSUrl?,
cavv?, PAResVerified?)>
<!ELEMENT type (#PCDATA)>
<!ELEMENT success (#PCDATA)>
<!ELEMENT message (#PCDATA)>
<!ELEMENT PaReq (#PCDATA)>
<!ELEMENT TermUrl (#PCDATA)>
<!ELEMENT MD (#PCDATA)>
<!ELEMENT ACSUrl (#PCDATA)>
<!ELEMENT cavv (#PCDATA)>
<!ELEMENT PAResVerified (#PCDATA)>
```

8. What Information will I get as a Response to My Transaction Request?

For each transaction you will receive a response message. For a full description of each field please refer to Appendix B. Definitions of Response Fields

To determine whether a transaction is successful or not the field that must be checked is ResponseCode. See the table below to determine the transaction result.

Response Code	Result
0 – 49 (inclusive)	Approved
50 – 999 (inclusive)	Declined
Null	Incomplete

For a full list of response codes and the associated message please refer to https://developer.moneris.com and download the Response Code document.

9. How Do I Test My Solution?

A testing environment is available for you to connect to while you are integrating your site to our payment gateway. The test environment is generally available 7x24, however since it is a test environment we cannot guarantee 100% availability. Also, please be aware that other merchants are using the test environment so you may see transactions and user IDs that you did not create. As a courtesy to others that are testing we ask that when you are processing Refunds, changing passwords and/or trying other functions that you use only the transactions/users that you created.

When using the APIs in the test environment you will need to use test store_id and api_token. These are different than your production IDs. The IDs that you can use in the test environment are in the table below.

NOTE For MasterPass test credentials, please email us at eproducts@moneris.com with subject as "Requesting test credentials for MasterPass".

When testing you may use the following test card numbers well with any future expiry date.

Test Card Numbers	
Card Plan	**Card Number**
MasterCard	5454545454545454
Visa	4242424242424242
Amex	373599005095005
Diners	36462462742008

To access the Merchant Resource Centre in the test environment go to https://esqa.moneris.com/mpg and use the logins provided in the Test ID table.

The test environment has been designed to replicate our production environment as closely as possible. One major difference is that we are unable to send test transactions onto the production authorization network and thus Issuer responses are simulated. Additionally, the requirement to emulate approval, decline and error situations dictates that we use certain transaction variables to initiate various response and error situations.

The test environment will approve and decline credit card transactions based on the penny value of the amount field. For example, a transaction made for the amount of $9.00 or $1.00 will approve since the .00 penny value is set to approve in the test environment. Transactions in the test environment should not exceed $11.00. This limit does not exist in the production environment. For a list of all current test environment responses for various penny values, please see the Test Environment Penny Response table as well as the Test Environment eFraud Response table, available at https://developer.moneris.com.

NOTE These responses may change without notice. Moneris Solutions recommends you regularly refer to our download website to check for possible changes.

10. What Do I Need to Include in the Receipt?

Visa and MasterCard expect certain variables be returned to the cardholder and presented as a receipt when a transaction is approved. These 12 fields are listed below. A sample receipt is provided in Appendix D. Sample Receipt.

1. Amount
2. Transaction Type
3. Date and Time
4. Auth Code
5. Response Code
6. ISO Code
7. Response Message
8. Reference Number
9. Goods and Services Order
10. Merchant Name
11. Merchant URL
12. Cardholder Name
13. Return Policy (only a requirement for e-commerce transactions)

11. How Do I Activate My Store?

Once you have received your activation letter/fax go to https://www3.moneris.com/connect/en/activate/index.php as instructed in the letter/fax. You will need to input your store ID and merchant ID then click on ‘Activate’. In this process you will need to create an administrator account that you will use to log into the Merchant Resource Centre to access and administer your eSELECTplus store. You will need to use the Store ID and API Token to send transactions through the API.

Once you have created your first Merchant Resource Centre user, please log on to the Interface by clicking the “eSELECTplus” button. Once you have logged in please proceed to ADMIN and then STORE SETTINGS. At the bottom please place a check beside the APIs that you are using. This will allow us to keep you up to date regarding any changes to the APIs that may affect your store. Also, this is where you may locate your production API Token.

12. How Do I Configure My Store For Production?

Once you have completed your testing you are ready to point your store to the production host. You will need to change the “host” to be www3.moneris.com. You will also need to change the store_id to reflect your production store ID and well the api_token must be changed to your production token to reflect the token that you received during activation.

Once you are in production you will access the Merchant Resource Centre at https://www3.moneris.com/mpg. You can use the store administrator ID you created during the activation process and then create additional users as needed.

For further information on how to use the Merchant Resource Centre please see the eSELECTplus Merchant Resource Centre User’s Guide which is available at https://developer.moneris.com.

13. How Do I Get Help?

If you require technical assistance while integrating your store, please contact the eSELECTplus Support Team:

For Technical Support (7/24):
Phone: 1-866-319-7450
Email: eselectplus@moneris.com

For Integration Support (M-F 8am-8pm EST):
Phone: 1-866-562-4354
Email: eproducts@moneris.com

When sending an email support request please be sure to include your name and phone number, a clear description of the problem as well as the type of API that you are using. **For security reasons, please do not send us your API Token combined with your store ID, or your merchant number and device number in the same email.**

14. Appendix A. Definition of Request Fields

Required Fields		
Variable Name	**Size/Type**	**Description**
accept		MIME types the browser accepts
amount	9 / decimal	Amount of the transaction. This must contain 3 digits with two penny values. The minimum value passed can be 0.01 and the maximum 9999999.99
avs_street_number	19 / an	Street Number & Street Name (max – 19 digit limit for street number and street name combined). This must match the address that the issuing bank has on file.
avs_street_name		
avs_zipcode	10 / an	Zip or Postal Code – This must match what the issuing banks has on file.
cavv		This is a value that is provided by the Moneris MPI or by a third party MPI. It is part of a VBV transaction.
checkout_resource_url	an	Endpoint used to request the users billing and shipping information from MasterPass.
crypt_type	1 / an	E-Commerce Indicator: 1 - Mail Order / Telephone Order - Single 2 - Mail Order / Telephone Order - Recurring 3 - Mail Order / Telephone Order - Instalment 4 - Mail Order / Telephone Order - Unknown Classification 5 - Authenticated E-commerce Transaction (VBV) 6 – Non Authenticated E-commerce Transaction (VBV) 7 - SSL enabled merchant 8 - Non Secure Transaction (Web or Email Based) 9 - SET non - Authenticated transaction
cust_id	50 / an	This is an optional field that can be sent as part of a Purchase or PreAuth request. It is searchable from the Moneris Merchant Resource Centre. It is commonly used for policy number, membership number, student ID or invoice number.
dynamic_descriptor	20/an	Merchant defined description sent on a per-transaction basis that will appear on the credit card statement. Dependent on the card Issuer, the statement will typically show the dynamic desciptor appended to the merchant's existing business name separated by the "/" character. Please note that the combined length of the merchant's business name, forward slash "/" character, and the dynamic descriptor may not exceed 22 characters. -Example- Existing Business Name: ABC Painting Dynamic Descriptor: Booking 12345 Cardholder Statement Displays: ABC Painting/Booking 1
MD	1024 / an	This is information that you would like echoed back in the response.
merchantUrl		This is the URL to which you would like the MPI response sent to.
mp_request_token		oauthtoken returned in response by MasterPass
oauthtoken	an	oauth_token is sent in the signature base string, authorization

		header and redirect URL
oauth_verifier	an	Verifier is returned on the callback and used in the access token request.
order_id	50 / an	Merchant defined unique transaction identifier - must be unique for every Purchase, Preauth and Independent Refund attempt. For Refunds, completion and purchasecorrection the order_id must reference the original transaction.
paypass_send_shopping_cart paypass_retrieve_checkout_data paypass_purchase paypass_preauth paypass_cavv_purchase paypass_cavv_preauth paypass_purchasecorrection paypass_refund paypass_completion paypass_txn		These transaction types are case sensitive field that must be all in lower case. See section 4 for description.
subtotal	9/ decimal	Amount of the transaction. This must contain 3 digits with two penny values. The minimum value passed can be 0.01 and the maximum 9999999.99
suppress_shipping_address	true/ false	Flag to suppress the shipping options presented to the user. This parameter is optional and will default to false.
txn_num	20/ an	Gateway Transaction identifier. Returned in paypass transaction response as TransID
UserAgent		The browser details

NOTE

The order_id allows the following characters **a-z A-Z 0-9 _ - : . @ spaces**

All other request fields allow the following characters: **a-z A-Z 0-9 _ - : . @ $ =** /

15. Appendix B. Definitions of Response Fields

<table>
<tr><th colspan="3">Response Fields</th></tr>
<tr><th>Variable Name</th><th>Size/Type</th><th>Description</th></tr>
<tr><td>ReceiptId</td><td>50 / an</td><td>order_id specified in request</td></tr>
<tr><td>ReferenceNum</td><td>18 / num</td><td>The reference number is an 18 character string that references the terminal used to process the transaction as well as the shift, batch and sequence number. This data is typically used to reference transactions on the host systems and must be displayed on any receipt presented to the customer. This information should be stored by the merchant. The following illustrates the breakdown of this field where "660123450010690030” is the reference number returned in the message, "66012345" is the terminal ID, "001" is the shift number, "069" is the batch number and "003" is the transaction number within the batch.

Moneris Host Transaction identifier.</td></tr>
<tr><td>ReponseCode</td><td>3 / num</td><td>Transaction Response Code
< 50: Transaction approved
>= 50: Transaction declined
NULL: Transaction was not sent for authorization

* If you would like further details on the response codes that are returned please see the Response Codes document available at https://developer.moneris.com.</td></tr>
<tr><td>ISO</td><td>2 / num</td><td>ISO response code</td></tr>
<tr><td>AuthCode</td><td>8 / an</td><td>Authorization code returned from the issuing institution</td></tr>
<tr><td>TransTime</td><td>##:##:##</td><td>Processing host time stamp</td></tr>
<tr><td>TransDate</td><td>yyyy-mm-dd</td><td>Processing host date stamp</td></tr>
<tr><td>TransType</td><td>an</td><td>Type of transaction that was performed</td></tr>
<tr><td>Complete</td><td>true/false</td><td>Transaction was sent to authorization host and a response was received</td></tr>
<tr><td>Message</td><td>100 / an</td><td>Response description returned from issuing institution.</td></tr>
<tr><td>TransAmount</td><td>9 / numeric</td><td>Amount authorized by gateway. Contains 2 decimals.</td></tr>
<tr><td>CardType</td><td>2 / an</td><td>Credit Card Type</td></tr>
<tr><td>TransID</td><td>20/ an</td><td>Gateway Transaction identifier used for follow-on transactions (void, refund or completion)</td></tr>
<tr><td>TimedOut</td><td>True/False</td><td>Transaction failed due to a process timing out</td></tr>
<tr><td>Ticket</td><td>n/a</td><td>reserved</td></tr>
<tr><td>MPRequestToken</td><td>an</td><td>MasterPass token for your transaction</td></tr>
<tr><td>MPRedirectUrl</td><td>an</td><td>MasterPass redirect Url for customer to login into their account</td></tr>
<tr><td>type</td><td>99 / an</td><td>VERes or PARes or error defines what type of response you are receiving</td></tr>
<tr><td>success</td><td>true/false</td><td>Returns whether the attempt was successful or not</td></tr>
<tr><td>message</td><td>alpha</td><td>Will contain:
<table>
<tr><th>Txn</th><th>Action</th></tr>
<tr><td>Y</td><td>Create VBV verification form popup window.</td></tr>
<tr><td>N</td><td>Send purchase or preauth with crypt type 6</td></tr>
<tr><td>U</td><td>Send purchase or preauth as crypt type 7</td></tr>
</table></td></tr>
</table>

ACS	Action
Y or A (<success>true</success>)	Proceed with cavv purchase or cavv preauth
N	Transaction must be cancelled, authentication failed.
U or time out	Send purchase or preauth as crypt type 7

PARes	n/a	Variable Length – data that Visa/MasterCard passes and needs for authentication
TermUrl	255 / an	The URL to which the PARes is returned. Please refer to section 9.d.
MD	1024 / an	Merchant defined data that will be echoed back
ACSUrl	255 / an	URL that will generate the inLine window
cavv	28 /an	Visa/MasterCard authentication field

MasterPass Response Fields

Variable Name	Size/Type	Description
CardBrandId	8 /an	Identifies the card brand id e.g. master for MasterCard
CardBrandName	255/ string	Identifies the card brand e.g. MasterCard
CardBillingAddressCity	25/ string	Cardholder's city
CardBillingAddressCountry	2 /string	Cardholder's country. Defined by ISO 3166- 1 alpha-2 digit country codes e.g. US is United States, AU is Australia, CA is Canada, GB is United Kingdom, etc.
CardBillingAddressCountrySubdivision	2 /string	Cardholder's country subdivision. Defined by ISO 3166-1 alpha-2 digit code e.g. US-VA is Virginia, US-OH is Ohio
CardBillingAddressLine1	40 /string	Address line 1 used for Street number and Street Name.
CardBillingAddressLine2	40 /string	Address line 2 used for Apt Number, Suite Number ,etc.
CardBillingAddressPostalCode	20 /string	Postal Code or Zip Code appended to mailing address for the purpose of sorting mail.
CardCardHolderName	100 /string	Cardholder name
CardExpiryMonth	2 /numeric	Expiration month displayed on the payment card.
CardExpiryYear	4 /numeric	Expiration year displayed on the payment card.
ContactEmailAddress	512/ string	Contact Email Address
ContactFirstName	20/ string	Contact First Name
ContactLastName	20/ string	Contact Surname
ContactPhoneNumber	20/ string	Contact Phone
ShippingAddressCity	25/ string	Cardholder's city
ShippingAddressCountry	2 /string	Cardholder's country. Defined by ISO 3166- 1 alpha-2 digit country codes e.g. US is United States, AU is Australia, CA is Canada, GB is United Kingdom, etc.

ShippingAddressCountrySubdivision	2 /string	Cardholder’s country subdivision. Defined by ISO 3166-1 alpha-2 digit code e.g. US-VA is Virginia, US-OH is Ohio
ShippingAddressLine1	40 /string	Address line 1 used for Street number and Street Name.
ShippingAddressLine2	40 /string	Address line 2 used for Apt Number, Suite Number ,etc.
ShippingAddressPostalCode	20 /string	Postal Code or Zip Code appended to mailing address for the purpose of sorting mail.
ShippingAddressRecipientName	100/ string	Name of person set to receive the shipped order.
ShippingAddressRecipientPhoneNumber	20/ string	Phone of the person set to receive the shipped order.
PayPassWalletIndicator	3 /string	The Wallet Indicator is returned by the wallet platform.
AuthenticationOptionsAuthenticateMethod	alpha	Method used to authenticate the cardholder at checkout. Valid values are MERCHANT ONLY and 3DS.
AuthenticationOptionsCardEnrollmentMethod	alpha-numeric	Method by which the card was added to the wallet. Valid values are: ▪ Manual ▪ Direct Provisioned ▪ 3DS Manual ▪ NFC Tap
AuthenticationOptionsCAvv	alpha-numeric	(CAVV) Cardholder Authentication Verification Value generated by card issuer upon successful authentication of the cardholder and which should be. This should be passed in the authorization message
AuthenticationOptionsEciFlag	alpha-numeric	Electronic commerce indicator (ECI) flag. Present when the PaRes value is "Y" or "A." Possible values are: 01-Indicates Merchant Liability (MasterCard) 02-Indicates Card Issuer Liability (MasterCard) 05-Indicates Card Issuer Liability (Visa) 06-Indicates Card Issuer Liability (Visa) - beware of fallback Cavv Result code of “B” 07-Indicates Merchant Liability (Visa)
AuthenticationOptionsPaResStatus	alpha-numeric	A message formatted, digitally signed, and sent from the ACS (issuer) to the MPI providing the results of the issuer’s SecureCode/Verified by Visa cardholder authentication. Possible values are: ▪ Y-the card was successfully authenticated via 3DS A-signifies that either: 1. The transaction was successfully authenticated via a 3DS attempts transaction 2. The cardholder was prompted to activate 3DS during shopping but declined (Visa). ▪ U-Authentication results were unavailable
AuthenticationOptionsSCEnrollmentStatus	alpha-numeric	SecureCode Enrollment Status: Indicates if the issuer of the card supports payer authentication for this card. Possible values are: ▪ Y-The card is eligible for 3DS authentication. ▪ N-The card is not eligible for 3DS authentication. ▪ U-Lookup of the card's 3DS eligibility status was

		either unavailable, or the card is inapplicable (i.e. prepaid cards).
AuthenticationOptionsSignatureVerification	alpha-numeric	Signature Verification. Possible values are: ▪ Y- Indicates that the signature of the PaRes has been validated successfully and the message contents can be trusted. ▪ N-Indicates that for a variety of reasons (tampering, certificate expiration, etc.) the PaRes could not be validated, and the result should not be trusted.
AuthenticationOptionsXID	alpha-numeric	Transaction identifier resulting from authentication processing.
CardAccountNumber	alpha-numeric	Masked Card number or primary account number that identifies the card (First 4 and last 4 digits only)

16. Appendix C. Error Messages

Global Error Receipt – You are not connecting to our servers. This can be caused by a firewall or your internet connection.

Response Code = NULL – The response code can be returned as null for a variety of reasons. A majority of the time the explanation is contained within the Message field. When a 'NULL' response is returned it can indicate that the Issuer, the credit card host, or the gateway is unavailable, either because they are offline or you are unable to connect to the internet. A 'NULL' can also be returned when a transaction message is improperly formatted.

Below are error messages that are returned in the Message field of the response.

Message: XML Parse Error in Request: <System specific detail>
Cause: For some reason an improper XML document was sent from the API to the servlet

Message: XML Parse Error in Response: <System specific detail>
Cause: For some reason an improper XML document was sent back from the servlet

Message: Transaction Not Completed Timed Out
Cause: Transaction times out before the host responds to the gateway

Message: Request was not allowed at this time
Cause: The host is disconnected

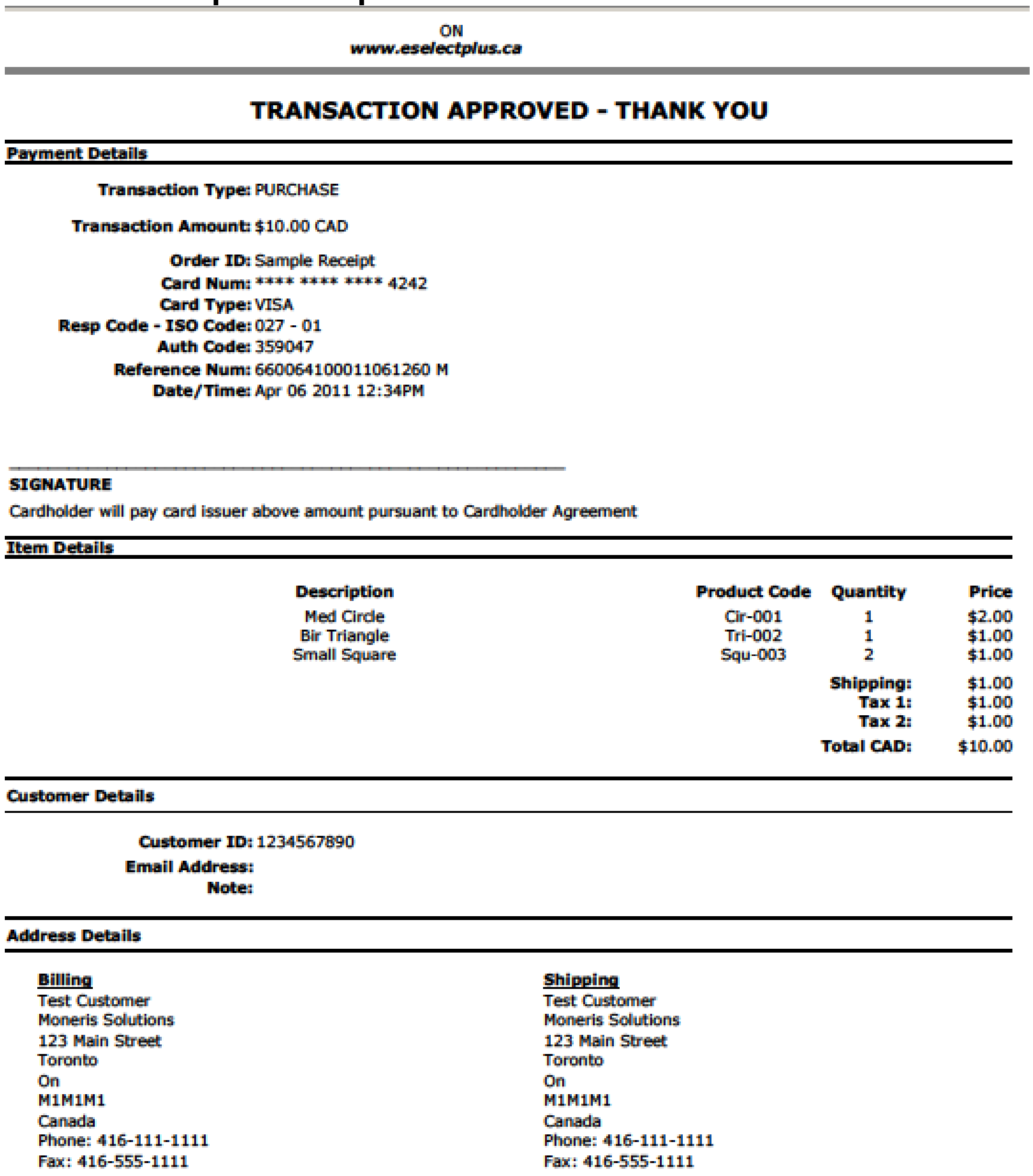
Message: Could not establish connection with the gateway: <System specific detail>
Cause: Gateway is not accepting transactions or server does not have proper access to internet

Message: Input/Output Error: <System specific detail>
Cause: Servlet is not running

Message: The transaction was not sent to the host because of a duplicate order id
Cause: Tried to use an order id which was already in use

Message: The transaction was not sent to the host because of a duplicate order id
Cause: Expiry Date was sent in the wrong format

17. Appendix D. Sample Receipt



ON
www.eselectplus.ca

TRANSACTION APPROVED - THANK YOU

Payment Details

Transaction Type:	PURCHASE
Transaction Amount:	$10.00 CAD
Order ID:	Sample Receipt
Card Num:	**** **** **** 4242
Card Type:	VISA
Resp Code - ISO Code:	027 - 01
Auth Code:	359047
Reference Num:	660064100011061260 M
Date/Time:	Apr 06 2011 12:34PM

SIGNATURE

Cardholder will pay card issuer above amount pursuant to Cardholder Agreement

Item Details

Description	Product Code	Quantity	Price
Med Circle	Cir-001	1	$2.00
Bir Triangle	Tri-002	1	$1.00
Small Square	Squ-003	2	$1.00
		Shipping:	$1.00
		Tax 1:	$1.00
		Tax 2:	$1.00
		Total CAD:	$10.00

Customer Details

Customer ID:	1234567890
Email Address:	
Note:	

Address Details

Billing
Test Customer
Moneris Solutions
123 Main Street
Toronto
On
M1M1M1
Canada
Phone: 416-111-1111
Fax: 416-555-1111

Shipping
Test Customer
Moneris Solutions
123 Main Street
Toronto
On
M1M1M1
Canada
Phone: 416-111-1111
Fax: 416-555-1111

eSELECTplus™

Copyright Notice

This document has been produced as a reference guide to assist Moneris client’s hereafter referred to as merchants. Every effort has been made to the make the information in this reference guide as accurate as possible. The authors of Moneris Solutions shall have neither liability nor responsibility to any person or entity with respect to any loss or damage in connection with or arising from the information contained in this reference guide.

Trademarks

Printed in Canada.

10 9 8 7 6 5 4 3 2 1