Merchant Integration Guide

CSV File Upload - v. 1.1.2

Moneris SOLUTIONS

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. Introduction

By definition, a batch file may be used to forward large groups of transactions for processing. Merchants may forward multiple transaction types, for multiple card plans in a single batch for processing, provided that the merchant is registered for the card plans included in the file.

The CSV Batch File option refers to the ability to upload batch files where the fields are presented in a comma delimited format.

There are a number of pieces of information that must be included in the batch file in a specific order. The following outlines the fields and the order in which they are expected for each transaction type. The last section of this document describes the information that will be provided back to you in the response file of processed batch files.

2. System and Skill Requirements

Before you start you will need to:

1. Have SFTP client software which must use SSH2
2. Create the Batch File which must have the ".csv" extension

Note: The CVD value supplied by the cardholder should simply be passed to the eSELECTplus payment gateway. Under no circumstances should it be stored for subsequent uses or displayed as part of the receipt information.

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

Sending your batch file:

1. Establish an SFTP session with Moneris' SFTP server host using your SFTP client software
2. Log in using your Batch account username and password (please refer to NOTE)
3. Upload the Batch File
4. Wait approximately 2 seconds per transaction + 30 minutes or longer for processing
5. Retrieve your response file from the /out directory

NOTE	To receive your Production SFTP Username and Password, please have your main contact person e-mail the eSELECTplus Integration Support department. E-mail: eselectplus@moneris.com When e-mailing, please provide your merchant name, store ID, as well as your business address, phone and fax number. Your SFTP account information will be sent via fax to the number provided.

4. Transaction Types and Transaction Flow

eSELECTplus supports a wide variety of transactions through Batch File Upload. Below is a list of transactions supported, other terms used for the transaction type are indicated in brackets.

Basic Transactions

Purchase – (sale) The Purchase transaction verifies funds on the customer's card, removes the funds and readies them for deposit into the merchant's account.

PreAuth – (authorisation / preauthorisation) The PreAuth verifies and locks funds on the customer's credit card. The funds are locked for a specified amount of time, based on the card issuer. To retrieve the funds from a PreAuth so that they may be settled in the merchant's account a Capture must be performed.

Capture – (Completion / PreAuth Completion) Once a PreAuth is obtained the funds that are locked need to be retrieved from the customer's credit card. The Capture retrieves the locked funds and readies them for settlement into the merchant's account.

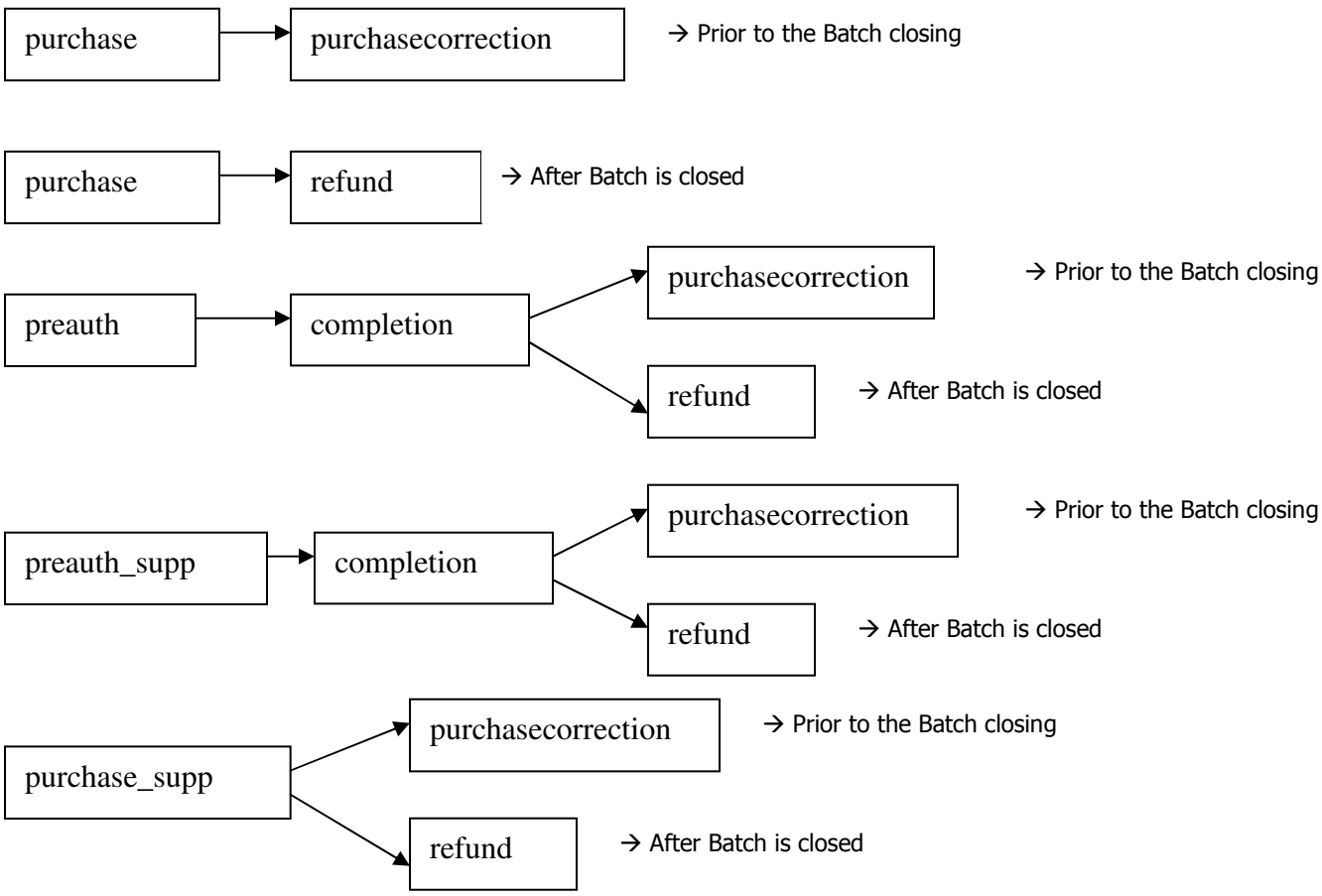
Void – (Correction / Purchase Correction) Purchases and Captures can be voided the same day* that they occur. A Void must be for the full amount of the transaction and will remove any record of it from the cardholder's statement.

Refund – (Credit) A Refund can be performed against a Purchase or a Capture to refund any part, or all of the transaction.

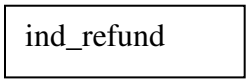
Independent Refund – (Credit) An Independent Refund can be performed to credit money to a Credit Card. This transaction does not require a prior Purchase or Capture.

* A Void can be performed against a transaction as long as the batch that contains the original transaction remains open.

Process Flow for Basic Transactions



Transactions with no Follow-on Required



Vault Admin Transactions

Add CC Vault Profile (res_add_cc) – Create a new credit card profile. The fields which may be sent in are outlined in the transaction examples which can be found in section 8 of this documentation.

Delete Vault Profile (res_delete) – Delete an existing profile of any payment type using the unique data_key which was assigned when the profile was first added. The fields which may be sent in are outlined in the transaction examples which can be found in section 8 of this documentation.

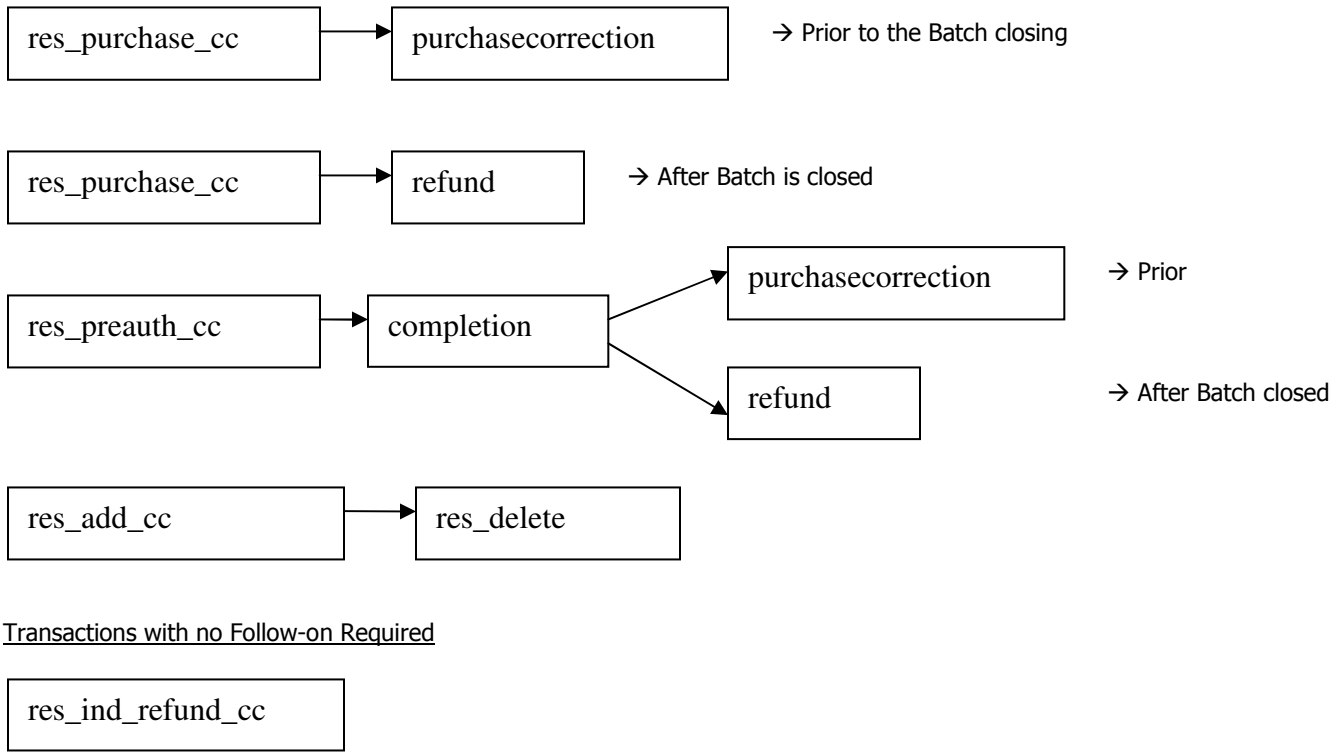
Vault Financial Transactions

Vault CC PreAuth (res_preauth_cc) – This is a Preauthorization transaction for Credit Card profiles only. This transaction will use a unique data_key which will identify a previously registered Credit Card profile. The details within the profile will be submitted to perform the Preauthorization transaction.

Vault CC Purchase (res_purchase_cc) - This is a Purchase transaction for Credit Card profiles only. This transaction will use a unique data_key which will identify a previously registered Credit Card profile. The details within the profile will be submitted to perform the Purchase transaction.

Vault CC Independent Refund (res_ind_refund_cc) - This is an independent refund transaction for Credit Card profiles only. This transaction will use a unique data_key which will identify a previously registered Credit Card profile. The details within the profile will be submitted to perform the refund transaction.

Process Flow for Vault Transactions



Transactions with no Follow-on Required

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. Please note that the fields must be in the correct order.

Required Fields	
Transaction Type	**Fields**
	Basic Transaction Types
Purchase	purchase (trans_type), order_id, amount, pan, exp_date, crypt_type, [recur – optional]
Purchase (with cust_id)	purchase_supp, order_id, amount, pan, exp_date, crypt_type, cust_id
Pre-authorization	preauth, order_id, amount, pan, exp_date, crypt_type
Pre-authorization (with cust_id)	preauth_supp, order_id, amount, pan, exp_date, crypt_type, cust_id
Capture	completion, order_id (from preauth), amount, txn_number (from preauth), crypt_type
Void	purchasecorrection, order_id (from orig txn), txn_number (from completion or purchase), crypt_type
Refund	refund, order_id(from orig txn), amount, txn_number(from completion/purchase),crypt_type
Independent Refund	ind_refund, order_id, amount, pan, exp_date, crypt_type
	Vault Initiated Financial Transactions (Basic)*
Vault CC Purchase	res_purchase_cc,order_id,cust_id,amount,data_key,crypt_type
Vault CC PreAuth	res_preauth_cc, order_id, cust_id, amount, data_key, crypt_type
Vault CC Ind Refund	res_ind_refund_cc, order_id, cust_id, amount, data_key, crypt_type

* Check appendix A for field definitions. Please refer to section 8 for the Vault Admin Transactions.

NOTE

The trans_type (ex. purchase or ind_refund) 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.

The format for exp_date is **YYMM**. Please note that this is different from what appears on the credit card.

6. Example Transaction

```
purchase,order_1_testing,13.00,4242424242424242,0304,1
purchase_supp,order_2_testing,2.00,4242424242424242,0908,1,customer_1
purchase,order_3_testing,13.00,5454545454545454,0403,1
preauth,order_4_testing,14.00,4242424242424242,0503,1
res_purchase_cc,mjr-res-purch-210510-251-1,moneris,1.00,M6eGTseSjlDBxYRICD3rgAhBn,1
res_preauth_cc,mjr-res-preauth-210510-251-1,moneris,1.00,Ef3QO7bzCE3hTzmDqjvC5dMdl,1
res_ind_refund_cc,mjr-res-indrefund-210510-251-1,moneris,1.00,dOeX9Hu7pRsLIJNcBphTTOmer,1
```

Corresponding Example Response

```
order_1_testing,660021810013368320,027,01,010460,11:04:27,2006-06-20,00,true,APPROVED          *
=,13.00,V,63790-832-0,false,,null

order_2_testing,660021810013368330,027,01,009494,11:04:28,2006-06-20,00,true,APPROVED          *
=,2.00,V,63791-833-0,false,,null

order_3_testing,660021810013368340,027,01,010461,11:04:30,2006-06-20,00,true,APPROVED          *
=,13.00,M,63792-834-0,false,,null

order_4_testing,660021810013368350,027,01,010462,11:04:31,2006-06-20,01,true,APPROVED          *
=,14.00,V,63793-835-0,false,,null

mjr-res-purch-210510-251-1,660035500012243850,027,01,073570,14:52:55,2010-05-21,00,true,APPROVED
*                   =,1.00,M,112072-0_7,false,,,M6eGTseSjlDBxYRICD3rgAhBn,cc,,,bob@smith.com,this is my
note,5454***5454,0812,1,,,,

mjr-res-preauth-210510-251-1,660035500012243860,027,01,073571,14:53:01,2010-05-21,01,true,APPROVED
*                   =,1.00,M,112073-0_7,false,,,Ef3QO7bzCE3hTzmDqjvC5dMdl,cc,,,bob@smith.com,this is my
note,5454***5454,0812,1,,,,

mjr-res-indrefund-210510-251-1,660035500012243870,027,01,740332,14:53:08,2010-05-21,04,true,APPROVED
*                   =,1.00,M,112074-0_7,false,,,dOeX9Hu7pRsLIJNcBphTTOmer,cc,,,bob@smith.com,this is my
note,5454***5454,0812,1,,,,
```

7. Recurring Transaction

You can also upload recurring transactions via the CSV file upload. When uploading a recurring transaction you will need to indicate the recurring amount, interval and start date and the number of time it is to recur. You can also optionally bill a different amount immediately.

The recurring request can only be sent as a purchase. The format of the request is as follows:

Transaction Type	Fields
Purchase (with recurring billing)	purchase (trans_type), order_id, amount(charge now), pan, exp_date, crypt_type, recur_unit, start_now, start_date, num_recurs, period, recur_amount

Field Definitions		
Field Name	**Size/Type**	**Description**
purchase (trans_type)		All recurs must be sent with trans_type as purchase. The trans_type is case sensitive and must be all in lower case.
order_id	50 / alpha	Alphanumeric – must be unique for every order
amount (charge now)	9 / decimal	When start_now is set to 'true' the amount field becomes the amount to be billed immediately. When start_now is set to 'false' the amount field should be the same as the recur_amount field. The amount can be a minimum of 0.01 up to a maximum of 9999999.99. It must contain 2 decimal places and no $ sign.
pan	20 / num	Credit Card number – no spaces, no dashes (numeric only)
exp_date	4 / num	**Format – YYMM. Please note, this is opposite from what appears on the card.**
crypt_type	1 / num	E-Commerce Indicator: 2 - Mail Order/Telephone Order - Recurring Please refer to appendix A for a list of all Crypt Types.
recur_unit	day, week, month	The unit that you wish to use as a basis for the Interval. This can be set as day, week or month. Then using the "period" field you can configure how many days, weeks, months between billing cycles.
period	0 – 999/num	This is the number of recur_units you wish to pass between billing cycles. Example : period = 3, recur_unit=month -> Card will be billed every 3 months. period = 4, recur_unit=weeks -> Card will be billed every 4 weeks. period = 45, recur_unit=day -> Card will be billed every 45 days. Please note that the total duration of the recurring billing transaction should not exceed 5-10 years in the future.
start_date	YYYY/MM/DD	This is the date on which the first charge will be billed. The value must be in the future. It cannot be the day on which the transaction is being sent. If the transaction is to be billed immediately the start_now feature must be set to true and the start_date should be set at the desired interval after today.
start_now	true / false	When a charge is to be made against the card immediately start_now should be set to 'true'. If the billing is to start in the future then this value is to be set to 'false'. When start_now is set to 'true' the amount to be billed immediately may differ from the recur amount billed on a regular basis thereafter.
recur_amount	9 / decimal	Amount of the recurring transaction. This must contain 3 digits with two penny values. The minimum value passed can be 0.01 and the maximum 9999999.99. This is the amount that will be billed on the start_date and every interval thereafter.
num_recurs	1 – 99 / num	The number of times to recur the transaction.

NOTE When completing the recurring billing portion for a monthly payment, please keep in mind that to prevent the shifting of recur bill dates, avoid setting the start_date for anything past the 28th of any given month. For example, all billing dates set for the 31st of May will shift and bill on the 30th in June and will then bill the cardholder on the 30th for every subsequent month.

Example

```
purchase,order_1_recurring,3.00,4242424242424242,0712,2,week,true,2006/10/30,4,2,1.00
purchase,order_2_recurring,4.00,4242424242424242,0803,2,day,false,2006/10/15,30,10,4.00
purchase,order_3_recurring,5.00,4242424242424242,0610,2,month,true,2006/11/11,12,1,1.00
```

Example 1 will result in the card being billed immediately for $3.00 and then starting on the 30th of Oct it will be billed $1.00 every 2 weeks, 4 times. Final result will be that the card is charged 5 times – once for 3.00, and four times for 1.00.

Example 2 will result in the card being billed $4.00 starting the 15th of Oct and every 30 days thereafter, for a total of 10 times.

Example 3 will result in the card being billed immediately for 5.00 and then starting on the 11th of Nov it will be billed 1.00 every month for 12 months. The card will be billed a total of 13 times.

Corresponding Example Response

```
order_1_recurring,660021810013368380,027,01,008598,11:28:28,2006-06-20,00,true,APPROVED          *
=: Recurring transaction successfully registered.,3.00,V,63796-838-0,false,,null,true

order_2_recurring,null,null,null,null,null,null,null,true,Recurring transaction successfully
registered.,null,null,null,false,,null,true

order_3_recurring,660021810013368390,027,01,010535,11:28:34,2006-06-20,00,true,APPROVED          *
=: Recurring transaction successfully registered.,5.00,V,63797-839-0,false,,null,true
```

8. Vault Admin Transactions

Included below is the sample code for the Administrative transactions. These Administrative transactions allow the user to perform such tasks as creating new Vault profiles and deleting existing profiles.

res_add_cc – creates a new credit card profile.

res_delete – Delete an existing profile using the unique data_key which was assigned when the profile was first added. *It is important to note that once a profile is deleted, the information which was saved within can no longer be retrieved.*

Transaction Type	Fields
res_add_cc	res_add_cc,order_id,cust_id,phone,email,note,pan,exp_date,crypt_type
res_delete	res_delete,order_id,data_key

Note: Please refer to Appendix A for field definitions.

Examples

```
res_add_cc,mjr-res-add-210510-251-1,moneris,phone,email,note,4242424242424242,1111,7
res_delete,mjr-res-delete-210510-251-1,E08TCqdXsiOhDey9YZDa7QDMa
```

Example 1 processes an Add CC Vault Profile transaction which registers the new credit card profile. In this example all of the optional profile details have been provided: cust_id, phone, email, note

Example 2 processes a Delete Vault Profile transaction which will completely remove the profile and its details.

Corresponding Example Response (for response format check section 10)

```
mjr-res-add-210510-251-1,null,001,null,null,14:52:49,2010-05-21,null,true,Successfully registered CC
details.,null,null,null,false,,,we3u5tak7ce4UPiy1xRnDqqju,cc,moneris,phone,email,note,4242***4242,1111,7,,
,,
```

```
mjr-res-delete-210510-251-1,null,001,null,null,14:53:13,2010-05-21,null,true,Successfully deleted CC
details.,null,null,null,false,,,E08TCqdXsiOhDey9YZDa7QDMa,cc,moneris,phone,email,note,4242***4242,1111,7,,
,,
```

9. How do I send a Transaction?

We have an SFTP account setup in our test environment that you can use to upload test data for processing. To upload the file you will need to connect to:

	Test Environment	Production
Host	esqa.moneris.com	www3.moneris.com
Port	22	22
Server Type	SFTP using SSH2	SFTP using SSH2
Username	store3	Provided by Moneris **(please refer to NOTE)**
Password	store3	Provided by Moneris **(please refer to NOTE)**

NOTE

To receive your Production SFTP Username and Password, please have your main contact person e-mail the eSELECTplus Integration Support department.

E-mail: eselectplus@moneris.com

When e-mailing, please provide your merchant name, store ID, as well as your business address, phone and fax number. Your SFTP account information will be sent via fax to the number provided.

10. 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. After the file is successfully processed go to the /out directory of the SFTP account and download the .out file. This will contain all the transaction responses. Please review the response file for all of the transactions.

The response format for non-Vault transactions is below:

order_id, ReferenceNum, ResponseCode, ISO, AuthCode, TransTime, TransDate, TransType, Complete, Message, TransAmount, CardType, txn_number, TimedOut, BankTotals, Ticket

```
example: order_1_testing,660021810013368320,027,01,010460,11:04:27,2006-06-20,00,true,APPROVED          *
=,13.00,V,63790-832-0,false,,null
```

The response format for Vault initiated financial transactions is below:

When a Vault transaction is successfully processed you will receive a response message made up of 2 parts. The first part is identical to all other non-Vault financial transactions (please refer above). The second part will follow directly after part 1 and will include all of the fields registered within the profile. The data that will be included in part 2 depends on what type of profile was registered and what data was initially added. Any field that is not stored in the profile will still be included in the response by having its position marked by the comma delimiter.

Part1
order_id, ReferenceNum, ResponseCode, ISO, AuthCode, TransTime, TransDate, TransType, Complete, Message, TransAmount, CardType, txn_number, TimedOut, BankTotals, Ticket

Part2

data_key, payment_type, cust_id, phone, email, note, pan, exp_date, crypt_type, reserved, reserved, reserved, reserved

```
mjr-res-purch-210510-251-1,660035500012243850,027,01,073570,14:52:55,2010-05-21,00,true,APPROVED
*                    =,1.00,M,112072-0_7,false,,,M6eGTseSjlDBxYRICD3rgAhBn,cc,,,bob@smith.com,this is my
note,5454***5454,0812,1,,,,

mjr-res-preauth-210510-251-1,660035500012243860,027,01,073571,14:53:01,2010-05-21,01,true,APPROVED
*                    =,1.00,M,112073-0_7,false,,,Ef3QO7bzCE3hTzmDqjvC5dMdl,cc,,,bob@smith.com,this is my
note,5454***5454,0812,1,,,,

mjr-res-indrefund-210510-251-1,660035500012243870,027,01,740332,14:53:08,2010-05-21,04,true,APPROVED
*                    =,1.00,M,112074-0_7,false,,,dOeX9Hu7pRsLIJNcBphTTOmer,cc,,,bob@smith.com,this is my
note,5454***5454,0812,1,,,,
```

The response format for Vault admin transactions is below:

order_id, reserved, ResponseCode, reserved, reserved, TransTime, TransDate, reserved, Complete, Message, reserved, reserved, reserved, TimedOut, reserved, reserved, data_key, payment_type, cust_id, phone, email, note, pan, exp_date, crypt_type, reserved, reserved, reserved, reserved

```
mjr-res-add-210510-251-1,null,001,null,null,14:52:49,2010-05-21,null,true,Successfully registered CC
details.,null,null,null,false,,,we3u5tak7ce4UPiy1xRnDqqju,cc,moneris,phone,email,note,4242***4242,1111,7,,
,,

mjr-res-delete-210510-251-1,null,001,null,null,14:53:13,2010-05-21,null,true,Successfully deleted CC
details.,null,null,null,false,,,E08TCqdXsiOhDey9YZDa7QDMa,cc,moneris,phone,email,note,4242***4242,1111,7,,
,,
```

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://www3.moneris.com/connect/en/documents/index.html and download the Response Code document.

11. Filename Conventions

When you are uploading a file the filename must conform to certain standards. The file name must be alphanumeric. It cannot contain any spaces. The extension must be ".csv" and it must be lowercase. Filenames that do not meet these requirements will not be processed.

When uploading a file it must be put in your default root directory. Do not put it in the processing directory – it will not get processed. When a file is complete the response file will be put in the /out folder. The filename will be identical to what was uploaded but will have an extension ".out"

Request File	Response File
testing_june_20.csv	testing_june_20.csv.out

12. How Do I Test My Solution?

A testing environment is available for you to connect to while you are testing uploading .csv files for processing. 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 batch files that you did not create. As a courtesy to others that are testing we ask that when you are processing Refunds, deleting files and/or other functions that you use only the transactions/files that you created.

To upload .csv batch files to the test environment, please refer to section 9. To verify that these transactions have been properly received and processed, please refer to the .out file as well as logging into the Merchant Resource Centre to view reports of your transactions. To access the Merchant Resource Centre in the test environment go to https://esqa.moneris.com/mpg and use the logins provided in the table below. If you have uploaded the batch file to store 3, please log into the Merchant Resource Centre into store3 to locate your transactions. These test store ID's, usernames and passwords are different than your production IDs.

Test IDs		
Store ID	**Username**	**Password**
store1	DemoUser	password
store2	DemoUser	password
store3	DemoUser	password

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

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

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 transactions based on the penny value of the amount field.
For example, a transaction made for the amount of $399.00 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 $1000.00. 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 available at https://www3.moneris.com/connect/en/documents/index.html.

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

13. 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.

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 Batch Upload. This will allow us to keep you up to date regarding any changes to the Batch Upload that may affect your store.

Next, please have your main contact person contact the eSELECTplus Integration Support department for your SFTP account username and password. The support team may be reached at 1-866-562-4354 or by e-mail at eselectplus@moneris.com. When e-mailing, please provide your merchant name, store ID, as well as your business address, phone and fax number. Your SFTP account information will be sent via fax to the number provided.

14. How Do I Get Help?

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

Phone: 1-866-562-4354

Email: eselectplus@moneris.com

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

15. Appendix A. Definitions of Required Fields

Required Fields		
Variable Name	**Size/Type**	**Description**
Required Fields **(for Basic Credit Card Transactions)**		
trans_type	an	The transaction type is a case sensitive field that must be all in lower case. The trans_type field must match one of these available transaction types: • purchase • purchase_supp • preauth • preauth_supp • completion • purchasecorrection • refund • ind_refund
order_id	50 / an	Merchant defined unique transaction identifier - must be unique for every purchase, preauth and ind_refund attempt. For refund, completion and purchasecorrection the order_id must reference the original transaction.
pan	20 / variable	Credit Card Number - no spaces or dashes. Most credit card numbers today are 16 digits in length but some 13 digits are still accepted by some issuers. This field has been intentionally expanded to 20 digits in consideration for future expansion and/or potential support of private label card ranges.
exp_date	4 / num	Expiry Date - format **YYMM** no spaces or slashes. PLEASE NOTE THAT THIS IS REVERSED FROM THE DATE DISPLAYED ON THE PHYSICAL CARD WHICH IS MMYY
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
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
txn_number	255 / varchar	Used when performing follow on transactions - this must be filled with the value that was returned as the Txn_number in the response of the original transaction. When performing a Capture this must reference the PreAuth. When performing a Refund or a Void this must reference the Capture or the Purchase.
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.

Vault Fields

trans_type	an	The transaction type is a case sensitive field that must be all in lower case. The trans_type field must match one of these available transaction types: • res_add_cc • res_delete • res_purchase_cc • res_preauth_cc • res_ind_refund_cc
data_key	50 / an	An alphanumeric identifier used in Vault transactions to uniquely identify a Vault profile. The data_key is generated by Moneris Solutions and returned to the merchant when the profile is first registered using the res_add_cc transaction.
order_id	50 / an	Merchant defined unique transaction identifier. In a Vault admin transaction (res_add_cc, res_delete) the Order ID is only used to match the request with the response. The Order ID will not be stored within the Moneris Merchant Resource Centre (MRC). In all Vault financial transactions (res_purchase_cc, res_preauth_cc, res_ind_refund) the Order ID must be unique and will be used to reference the transaction within the MRC. Characters allowed for Order ID: **a-z A-Z 0-9 _ - : . @ spaces**
cust_id	50 / an	This is an optional field that can be either registered in a profile or sent as part of a res_purchase_cc, res_preauth_cc, or res_ind_refund_cc request. It is searchable from the Moneris Merchant Resource Centre. It is commonly used for policy number, membership number, student ID or invoice number.
phone	30 / an	Phone number of the customer. This is an optional field which can be sent in when creating a Vault profile.
email	30 / an	Email of the customer. This is an optional field which can be sent in when creating a Vault profile.
note	30 / an	This field can be used for supplementary information which is to be sent in with the transaction. This is an optional field which can be sent in when creating a Vault profile.

16. Appendix B. Definitions of Response Fields

Response Fields		
Variable Name	**Size/Type**	**Description**
order_id	50 / an	order_id specified in request
ReferenceNum	18 / num	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.
ReponseCode	3 / num	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://www3.moneris.com/connect/en/documents/index.html
ISO	2 / num	ISO response code
AuthCode	8 / an	Authorization code returned from the issuing institution
TransTime	##:##:##	Processing host time stamp
TransDate	yyyy-mm-dd	Processing host date stamp
TransType	2 / num	Type of transaction that was performed 00 = Purchase, Vault CC Purchase 01 = Preauthorization, Vault CC Preauthorization 02 = Capture 04 = Refund, Independent Refund, Vault CC Ind Refund 11 = Void
Complete	true/false	Transaction was sent to authorization host and a response was received
Message	100 / an	Response description returned from issuing institution.
TransAmount		
CardType	2 / alpha	Credit Card Type M = MasterCard V = Visa AX = American Express DC = Diners Card NO = Novus / Discover SE = Sears
Txn_number	20 / an	Gateway Transaction identifier
TimedOut	true/false	Transaction failed due to a process timing out
Ticket	n/a	reserved
RecurSucess	true/false	Indicates whether the transaction successfully registered.

17. Appendix C. Definitions of Response Fields for Vault Transactions

Response Fields		
Variable Name	**Size/Type**	**Description**
order_id	50 / an	order_id specified in request
reserved	N/A	Future use
ReponseCode	3 / num	Transaction Response Code < 50: Transaction approved >= 50: Transaction declined Vault Admin Responses (i.e. res_add_cc or res_delete) 001 Successfully registered CC details. Successfully deleted CC details. 983 Can not find previous 986 Incomplete: timed out 987 Invalid transaction Null Error: Malformed XML
TransTime	##:##:##	Processing host time stamp (24 hour clock)
TransDate	yyyy-mm-dd	Processing host date stamp
Complete	true / false	Transaction was sent to authorization host and a response was received
TimedOut	true / false	Transaction failed due to a process timing out
data_key	50 / an	The data_key specified in request. Or, in the case of a res_add_cc transaction it will specify the data_key created by Moneris Solutions.
payment_type	3 / an	Indicates the payment type associated with the Vault profile. Possible values: cc – credit card profile
cust_id	50 / an	The value stored in the vault, it will be blank if no data is stored.
phone	30 / an	The value stored in the vault, it will be blank if no data is stored.
email	30 / an	The value stored in the vault, it will be blank if no data is stored.
note	30 / an	The value stored in the vault, it will be blank if no data is stored.
pan	an	The first 4 and last 4 of the credit card number.
exp_date	4 / num	The credit card expiry date.
crypt_type	1 / an	The E-commerce Indicator.