



Merchant Integration Guide

Level 2/3 XML File Upload - v. 1.3.3



Revision Number	Date	Description
V1.3.0	March, 2014	Initial Release
V1.3.1	April, 2014	Updated all sections to reference the updated DTD specifications
V1.3.2	March 2015	Section 5 - Note added regarding non-purchasing card processing Appendix A <ul style="list-style-type: none"> - All amount fields' limits changed to 999999.99 - All amount fields indicate <u>not</u> to include a \$ sign - item_other_tax_type field options changed to 3 character tax type
V1.3.3	March 11 th , 2016	Added the 98 line item limitation for level23 over batch upload

Table of Contents

1.	Introduction	5
2.	System and Skill Requirements.....	5
3.	What is the Process I will need to follow?	5
4.	Transaction Types and Transaction Flow	6
5.	What Information Do I Need to Include in a Transaction Request?	8
6.	What Information Do I receive in Transaction Response?	9
7.	Filename Conventions	10
8.	How Do I Test My Solution?	10
9.	How Do I Activate My Store?.....	12
10.	How Do I Get Help?.....	12
11.	Appendix A: Definitions of Required Fields	13
12.	Appendix B: Definitions of Response Fields	16
13.	Appendix C: Error Messages.....	18
14.	Appendix D: XML Request DTD.....	19
15.	Appendix E: XML Response DTD	20
16.	Appendix F: XML Request Example	22
17.	Appendix G: XML Response XML.....	24

****** 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 XML Batch File option refers to the ability to upload batch files where the fields are presented in an XML 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 an SFTP client software which must use SSH2
2. Store ID and ApiToken of your account
3. Create the Batch File - it must have the ".l23.xml" extension and no spaces in its name.

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

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



NOTE

E-mail: eproducts@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.



If you looking to send non-level 2/3 transactions to process via XML batch upload process, please refer to our regular XML batch upload guide.

NOTE

You may also process corporate AMEX cards using the level 2/3 transactions below and receive funds for it, but level 2/3 data will not be posted to AMEX.

Level 2/3 Transactions

Level 2/3 Purchase (*batch_l23purchase* - sales) The Purchase transaction verifies funds on the customer's card, removes the funds and readies them for deposit into the merchant's account. It also submits level 2/3 detail sent along with the transaction request.

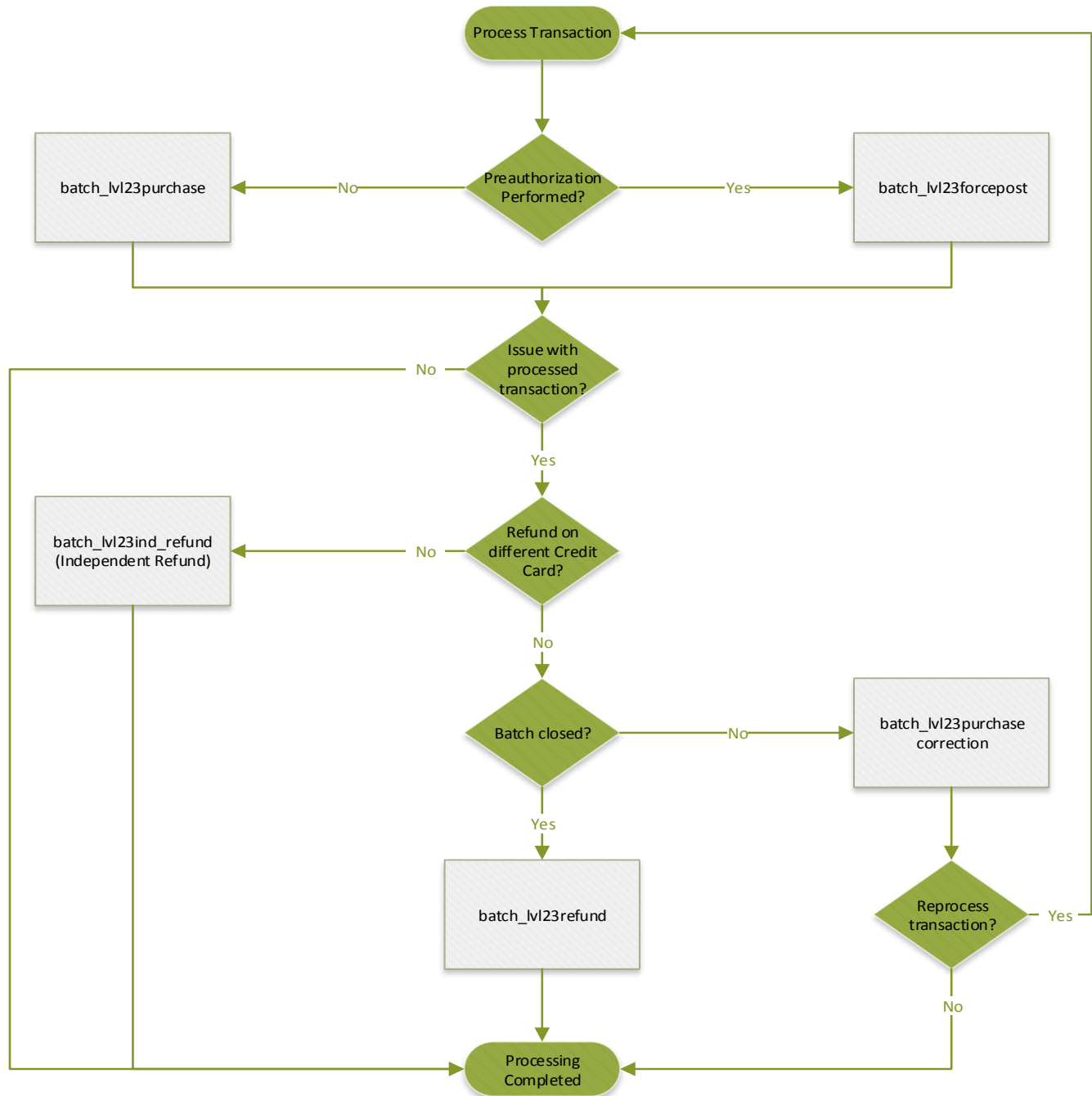
Level 2/3 Forcepost – (*batch_l23forcepost* - Capture / Completion / PreAuth Completion) Once a PreAuth is obtained the funds that are locked need to be retrieved from the customer's credit card. The Forcepost retrieves the locked funds from a previously processed PreAuth and readies them for settlement into the merchant's account along with submissions of level 2/3 details that is sent with the transaction request.

Level 2/3 Purchase Correction – (*batch_l23purchasecorrection* - Correction / Void) Purchases and Forcepost 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. This transaction type does not accept level 2/3 details as no information is submitted to bank.

Level 2/3 Refund – (*batch_l23refund* - Credit) A Refund with level 2/3 details can be performed against a Purchase or a Forcepost to refund any part, or all of the transaction. This refund will show up on cardholder's statement along with level 2/3 details that were submitted with this transaction request. A prior Purchase or Forcepost transaction is required to use this transaction.

Level 2/3 Independent Refund – (*batch_l23ind_refund* - Credit) An Independent Refund can be performed to credit money to a Credit Card along with the submission of level 2/3 details pertaining to this refund. This transaction does not require a prior Purchase or Forcepost.

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

Transaction Flow for level 2/3 batch upload process:

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 tables illustrates the fields that you can send for each transaction type. For a full description of each field, please refer to Appendix A: Definitions of Required Fields. XML Document Type Definition (DTD) has also been provided in Appendix D: XML Request DTD. Please note that the fields must be in the correct order and the field names must be exactly the same. For an example of all of the transaction types and their respective structures, please refer to Appendix F: XML Request Example.

Required Request Fields	
Super Root	Root
I23_batch	request (one or more)

Required Request Fields	
Root	Child
request	store_id, api_token, (batch_I23purchase batch_I23refund batch_I23ind_refund batch_I23purchasecorrection batch_I23forcepost)+

Required Transaction Fields	
Child	SubChild
batch_I23purchase	order_id, cust_id (optional), amount, pan, expdate, crypt_type, addendum1, addendum2 (one or more)
batch_I23forcepost	order_id, cust_id(optional), amount, pan, expdate, auth_code, crypt_type, addendum1, addendum2 (one or more)
batch_I23purchasecorrection	order_id, txn_number, crypt_type
batch_I23refund	order_id, amount, txn_number, crypt_type, addendum1, addendum2 (one or more)
batch_I23ind_refund	order_id, cust_id(optional), amount, pan, expdate, crypt_type, addendum1, addendum2 (one or more)

Required Level 2 Fields	
SubChild	Sub-SubChild
appendum1	customer_code, discount_amount, freight_amount, duty_amount, national_tax_amount, other_tax_amount, merchant_vat_no, customer_vat_no, destination_zip, ship_from_zip

Required Level 3 Fields	
SubChild	Sub-SubChild
appendum2	item_description, product_code, quantity, unit_cost, ext_amount, uom, item_discount_amount, item_tax_amount, item_other_tax_amount, item_other_tax_type



NOTE

The above mentioned transactions can only be performed on purchasing corporate card. Appendum1 is mandatory. Also at least 1 or more Appendum2 is required. If account is setup with Level2/3 Addendum Data bypass account rights, merchant may send in Non Purchasing Corporate Cards with addendum 1 and 2. In this scenario, transaction will be processed as a Level 1 transaction but the entire request XML including addendum 1 and 2 will be echoed back in the transaction response.

**NOTE**

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 999999.99.

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

6. What Information Do I receive in Transaction Response?

For each transaction, you will receive the original request along with response message. The following table shows the fields that you will receive. For a full description of each field please refer to Appendix B: Definitions of Response Fields. XML Document Type Definition (DTD) has also been provided in Appendix E: XML Response DTD. After the file is successfully processed, you can 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.

Required Request Fields	
Super Root	Root
I23_batch	response (one or more)

Response Fields	
Root	Child
response	orig_request, receipt

Original Request Fields	
Child	SubChild
orig_request	request NOTE: For more details on this SubChild, see request fields in section 5.
receipt	ReceiptId, ReferenceNum, ResponseCode, ISO, AuthCode, TransTime, TransDate, TransType, Complete, Message, TransAmount, CardType, TransID, TimedOut, BankTotals, Ticket, CorporateCard, MessageId

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.

7. 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 “.l23.xml” 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_mar_20.l23.xml	testing_mar_20.l23.xml.out

8. 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 batch files that you did not create. As a courtesy to others that are testing we ask that when you are processing Refunds, downloading/deleting batch files and/or trying other functions, that you use only the transactions/files that you created.

When using the XML Batch file upload 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.

Test IDs			
store_id	api_token	Username	Password
moneris	hurgle	DemoUser	password

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

Test Card Numbers	
Card Plan	Card Number
Corporate Visa	4242-424254545454
Corporate MasterCard	5454-545442424242
Corporate AMEX	3732-69005095005

To upload the test file you will need to connect to:

SFTP Accounts		
	Test Environment	Production
Host	esqa.moneris.com	www3.moneris.com
Port	22	22
Server Type	SFTP using SSH2	SFTP using SSH2
Username	moneris	Provided by Moneris (please refer to note in section 9)
Password	moneris	Provided by Moneris (please refer to note in 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 “moneris”, please log into the Merchant Resource Centre into “moneris” to locate your transactions. These test store ID’s, usernames and passwords are different than your production IDs.

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://developer.moneris.com>.



NOTE

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

9. How Do I Activate My Store?

Once you have received your activation letter/fax go to Moneris Merchant Resource Center at <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 this Store ID and API Token to send transactions via the SFTP Batch File Upload.

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. Also, this is where you may locate your production API Token.

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

**NOTE**

E-mail: eproducts@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. How Do I Get Help?

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

For Technical Support:

Phone: 1-866-319-7450 (Technical Difficulties)

Email: eselectplus@moneris.com

For Integration Support:

Phone: 1-866-562-4354

Email: eproducts@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 API Token combined with your store ID, or your merchant number and device number in the same email.**

11. Appendix A: Definitions of Required Fields

Request Field Details		
Variable Name	Size/Type	Description
store_id	10 / an	Store ID is provided to you in your eSELECTplus activation letter.
api_token	20 / an	API Token is obtained by logging to your eSELECTplus account and then navigating to [Admin] → [store settings].
batch_I23purchase batch_I23forcepost batch_I23purchasecorrection batch_I23refund batch_I23ind_refund	1+	This field defines the type of transaction to be performed.

Transaction Field Details		
Variable Name	Size/Type	Description
order_id	50 / an	Merchant defined unique transaction identifier - must be unique for every purchase, forcepost and ind_refund attempt. For refund and purchase correction the order_id must reference the original transaction.
cust_id	50 / an	This is an optional field that can be sent as part of a transaction request. It is searchable from the Moneris Merchant Resource Centre. It is commonly used for policy number, membership number, student ID or invoice number.
amount	14 / decimal	Amount of the transaction. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> Minimum value: 0.01 Maximum value: 999999.99
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.
expdate	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 MMY
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 TransID in the response of the original transaction. When performing a

		PurchaseCorrection or Refund, this must reference to response from the Purchase or Forcepost.
appendum1	1	This sub-subchild will contain level 2 transactions details. This field is mandatory for level 2/3 transactions.
appendum2	1+	This sub-subchild will contain level 3 transaction details and at least 1 appendum2 is required for level 2/3 transactions. You can send more than one appendum2 if needed.

Level 2 Field Details		
Variable Name	Size/Type	Description
customer_code	40 / an	A control number, such as purchase order number, project number, department allocation number or name that the purchaser supplied the merchant. Left-justified; may be spaces. <ul style="list-style-type: none"> Max size for Visa is 17 Max size for MasterCard is 25 Max size for AMEX is 40
discount_amount	14 / decimal	Discount on the entire transaction. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> Minimum value: 0.00 Maximum value: 999999.99
freight_amount	14 / decimal	Shipping cost on the entire transaction. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> Minimum value: 0.00 Maximum value: 999999.99
duty_amount	14 / decimal	Applicable duty fees on the entire transaction. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> Minimum value: 0.00 Maximum value: 999999.99
national_tax_amount	14 / decimal	National tax amount (GST/ HST) on the entire transaction. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> Minimum value: 0.00 Maximum value: 999999.99
other_tax_amount	14 / decimal	Provincial tax amount (PST/ QST) on the entire transaction. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> Minimum value: 0.00 Maximum value: 999999.99
merchant_vat_no	15 / an	Contains the merchant's VAT registration number or Single Business Reference Number (SBRN). For Canada, it is GST Number.
customer_vat_no	13 / an	Contains the customer's value-added tax registration number.
destination_zip	10 / an	Postal code of location to where goods are to be shipped
ship_from_zip	10 / an	Postal code of shipping location

Level 3 Field Details

IMPORTANT: For level3 fields sent over the batch method, there is a limit of 98 items per transaction.

Variable Name	Size/Type	Description
item_description	40 / an	Item description. <ul style="list-style-type: none"> • Max size for Visa is 26 • Max size for MasterCard is 35 • Max size for AMEX is 40
product_code	12 / an	Product or UPC code of the item.
quantity	14 / decimal	Number of same items purchased. <ul style="list-style-type: none"> • Minimum value: 1 • Maximum value: 999999999.9999
unit_cost	14 / decimal	Unit cost of item. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> • Minimum value: 0.00 • Maximum value: 999999.99
ext_amount	14 / decimal	Total price of the item [quantity * unit_cost]. It may include tax amount. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> • Minimum value: 0.00 • Maximum value: 999999.99
uom	12 / an	Item unit of measure. Please check the codes for Units of Measurement used in international trade.
item_discount_amount	14 / decimal	Discount amount applied on the total price of item. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> • Minimum value: 0.00 • Maximum value: 999999.99
item_tax_amount	14 / decimal	National tax amount on the item. For Canada, it is GST or HST. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> • Minimum value: 0.00 • Maximum value: 999999.99
item_other_tax_amount	14 / decimal	Any other type of tax amount on the item. It must contain 2 decimal places. Please <u>do not</u> include the \$ sign. <ul style="list-style-type: none"> • Minimum value: 0.00 • Maximum value: 999999.99
item_other_tax_type	4 / an	Indicator to determine the type of other tax applied. Possible values are: <ul style="list-style-type: none"> • PST – Provincial Sales Tax • QST – State/Provincial Tax ie. Quebec Sales Tax

12. Appendix B: Definitions of Response Fields

Response Fields		
Variable Name	Size/Type	Description
orig_request	echo	Everything contained in request root of request batch file is echoed back. All fields except pan follows the same format as shown in the request fields definition. pan field will be masked and you will receive only first 4 and last 4 digits of the card.
ReceiptId	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.
ReponseCode	3 / num	Moneris Host Transaction identifier. 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
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	an	Type of transaction that was performed 00 = Purchase 01 = Preauthorization 02 = Forcepost / Capture 04 = Refund, Independent 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	14 / decimal	Amount the transaction is processed for.
CardType	2 / alpha	Credit Card Type M = MasterCard V = Visa AX = American Express
TransID	20 / an	Gateway Transaction identifier. This field is referred in follow-on transaction as txn_number.

TimedOut	true/false	Transaction failed due to a process timing out
BankTotals	n/a	reserved for future use
Ticket	n/a	reserved for future use
CorporateCard	true/false	States whether the card is corporate or not
MessageId	20 / an	Processing Host indicator. Reserved.

13. 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

14. Appendix D: XML Request DTD

```
<!-- level23 batch request DTD -->
```

```
<!ELEMENT l23_batch (request)+>
```

```
<!ELEMENT request (store_id, api_token, (batch_l23purchase | batch_l23refund |
batch_l23ind_refund | batch_l23purchasecorrection| batch_l23forcepost)+)>
```

```
<!ELEMENT batch_l23purchase (order_id, cust_id?, amount, pan, expdate,
crypt_type, addendum1, addendum2+)>
```

```
<!ELEMENT batch_l23refund (order_id, amount, txn_number, crypt_type, addendum1,
addendum2+)>
```

```
<!ELEMENT batch_l23ind_refund (order_id, cust_id?, amount, pan, expdate,
crypt_type, addendum1, addendum2+)>
```

```
<!ELEMENT batch_l23purchasecorrection (order_id, txn_number, crypt_type)>
```

```
<!ELEMENT batch_l23forcepost (order_id, cust_id?, amount, pan, expdate,
auth_code, crypt_type, addendum1, addendum2+)>
```

```
<!ELEMENT order_id (#PCDATA)>
```

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

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

```
<!ELEMENT txn_number (#PCDATA)>
```

```
<!ELEMENT amount (#PCDATA)>
```

```
<!ELEMENT crypt_type (#PCDATA)>
```

```
<!ELEMENT expdate (#PCDATA)>
```

```
<!ELEMENT pan (#PCDATA)>
```

```
<!ELEMENT auth_code (#PCDATA)>
```

```
<!ELEMENT cust_id (#PCDATA)>
```

```
<!ELEMENT addendum1 (customer_code, discount_amount, freight_amount, duty_amount,
national_tax_amount, other_tax_amount, merchant_vat_no, customer_vat_no,
destination_zip, ship_from_zip)>
```

```
<!ELEMENT customer_code (#PCDATA)>
```

```
<!ELEMENT discount_amount (#PCDATA)>
```

```
<!ELEMENT freight_amount (#PCDATA)>
```

```
<!ELEMENT duty_amount (#PCDATA)>
```

```
<!ELEMENT national_tax_amount (#PCDATA)>
```

```
<!ELEMENT other_tax_amount (#PCDATA)>
```

```
<!ELEMENT merchant_vat_no (#PCDATA)>
```

```
<!ELEMENT customer_vat_no (#PCDATA)>
```

```
<!ELEMENT destination_zip (#PCDATA)>
```

```
<!ELEMENT ship_from_zip (#PCDATA)>
```

```
<!ELEMENT addendum2 (item_description,product_code, quantity, unit_cost,
ext_amount,uom, item_discount_amount, item_tax_amount, item_other_tax_amount,
item_other_tax_type>
```

```
<!-- Level 3 Multiple Recs. (Line Item Details)-->
```

```
<!ELEMENT item_description (#PCDATA)>
```

```
<!ELEMENT product_code (#PCDATA)>
```

```
<!ELEMENT quantity (#PCDATA)>
```

```
<!ELEMENT unit_cost (#PCDATA)>
```

```
<!ELEMENT ext_amount (#PCDATA)>
```

```
<!ELEMENT uom (#PCDATA)>
```

```
<!ELEMENT item_discount_amount (#PCDATA)>
```

```
<!ELEMENT item_tax_amount (#PCDATA)>
```

```
<!ELEMENT item_other_tax_amount (#PCDATA)>
```

```
<!ELEMENT item_other_tax_type (#PCDATA)>
```

15. Appendix E: XML Response DTD

```

<!-- level23 batch response DTD -->

<!ELEMENT l23_batch (response)+>

<!ELEMENT response (orig_request, receipt)>

<!ELEMENT orig_request (request)>

<!ELEMENT request (store_id, api_token, (batch_l23purchase | batch_l23refund |
batch_l23ind_refund | batch_l23purchasecorrection| batch_l23forcepost))>

<!ELEMENT receipt (ReceiptId, ReferenceNum, ResponseCode, ISO, AuthCode,
TransTime, TransDate, TransType, Complete, Message, TransAmount, CardType,
TransID, TimedOut, BankTotals, Ticket, CorporateCard, MessageId)>
<!ELEMENT ReceiptId (#PCDATA)>
<!ELEMENT ReferenceNum (#PCDATA)>
<!ELEMENT ResponseCode (#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 BankTotals (#PCDATA)>
<!ELEMENT Ticket (#PCDATA)>
<!ELEMENT ISO (#PCDATA)>
<!ELEMENT CorporateCard (#PCDATA)>
<!ELEMENT MessageId (#PCDATA)>

<!ELEMENT batch_l23purchase (order_id, cust_id?, amount, pan, expdate,
crypt_type, addendum1, addendum2+)>
<!ELEMENT batch_l23refund (order_id, amount, txn_number, crypt_type, addendum1,
addendum2+)>
<!ELEMENT batch_l23ind_refund (order_id, cust_id?, amount, pan, expdate,
crypt_type, addendum1, addendum2+)>
<!ELEMENT batch_l23purchasecorrection (order_id, txn_number, crypt_type)>
<!ELEMENT batch_l23forcepost (order_id, cust_id?, amount, pan, expdate,
auth_code, crypt_type, addendum1, addendum2+)>
<!ELEMENT order_id (#PCDATA)>
<!ELEMENT store_id (#PCDATA)>
<!ELEMENT api_token (#PCDATA)>
<!ELEMENT txn_number (#PCDATA)>
<!ELEMENT amount (#PCDATA)>
<!ELEMENT crypt_type (#PCDATA)>
<!ELEMENT expdate (#PCDATA)>
<!ELEMENT pan (#PCDATA)>
<!ELEMENT auth_code (#PCDATA)>
<!ELEMENT cust_id (#PCDATA)>

```

```
<!--ELEMENT addendum1 (customer_code, discount_amount, freight_amount, duty_amount,
national_tax_amount, other_tax_amount, merchant_vat_no, customer_vat_no,
destination_zip, ship_from_zip)-->
<!--ELEMENT customer_code (#PCDATA)-->
<!--ELEMENT discount_amount (#PCDATA)-->
<!--ELEMENT freight_amount (#PCDATA)-->
<!--ELEMENT duty_amount (#PCDATA)-->
<!--ELEMENT national_tax_amount (#PCDATA)-->
<!--ELEMENT other_tax_amount (#PCDATA)-->
<!--ELEMENT merchant_vat_no (#PCDATA)-->
<!--ELEMENT customer_vat_no (#PCDATA)-->
<!--ELEMENT destination_zip (#PCDATA)-->
<!--ELEMENT ship_from_zip (#PCDATA)-->

<!--ELEMENT addendum2 (item_description,product_code, quantity, unit_cost,
ext_amount,uom, item_discount_amount, item_tax_amount, item_other_tax_amount,
item_other_tax_type-->
<!-- Level 3 Multiple Recs. (Line Item Details)-->

<!--ELEMENT item_description (#PCDATA)-->
<!--ELEMENT product_code (#PCDATA)-->
<!--ELEMENT quantity (#PCDATA)-->
<!--ELEMENT unit_cost (#PCDATA)-->
<!--ELEMENT ext_amount (#PCDATA)-->
<!--ELEMENT uom (#PCDATA)-->
<!--ELEMENT item_discount_amount (#PCDATA)-->
<!--ELEMENT item_tax_amount (#PCDATA)-->
<!--ELEMENT item_other_tax_amount (#PCDATA)-->
<!--ELEMENT item_other_tax_type (#PCDATA)-->
```

16. Appendix F: XML Request Example

```

<?xml version="1.0" encoding="iso-8859-1"?>
<l23_batch>
  <request>
    <store_id>moneris</store_id>
    <api_token>hurgle</api_token>
    <batch_l23purchase>
      <order_id>NQA_20140428_0</order_id>
      <cust_id>custid123</cust_id>
      <amount>1.00</amount>
      <pan>4242-424254545454</pan>
      <expdate>1512</expdate>
      <crypt_type>7</crypt_type>
      <addendum1>
        <customer_code>customer_code</customer_code>
        <discount_amount>1.12</discount_amount>
        <freight_amount>2.23</freight_amount>
        <duty_amount>3.34</duty_amount>
        <national_tax_amount>4.56</national_tax_amount>
        <other_tax_amount>5.67</other_tax_amount>
        <merchant_vat_no>merchant_vat_no</merchant_vat_no>
        <customer_vat_no>cus_vat_no</customer_vat_no>
        <destination_zip>M1R 1W4</destination_zip>
        <ship_from_zip>M6K 2H9</ship_from_zip>
      </addendum1>
      <addendum2>
        <item_description>item_description</item_description>
        <product_code>product_code</product_code>
        <quantity>12</quantity>
        <unit_cost>1.2345</unit_cost>
        <ext_amount>2.89</ext_amount>
        <uom>EA</uom>
        <item_discount_amount>6.78</item_discount_amount>
        <item_tax_amount>4.32</item_tax_amount>
        <item_other_tax_amount>1.94</item_other_tax_amount>
        <item_other_tax_type>QST</item_other_tax_type>
      </addendum2>
    </batch_l23purchase>
  </request>
  <request>
    <store_id>moneris</store_id>
    <api_token>hurgle</api_token>
    <batch_l23ind_refund>
      <order_id>NQA_20140428_1</order_id>
      <cust_id>custid123</cust_id>
      <amount>1.00</amount>
      <pan>4242-424254545454</pan>
      <expdate>1512</expdate>
      <crypt_type>7</crypt_type>
      <addendum1>
        <customer_code>customer_code</customer_code>
        <discount_amount>1.12</discount_amount>
        <freight_amount>2.23</freight_amount>
        <duty_amount>3.34</duty_amount>
        <national_tax_amount>4.56</national_tax_amount>
        <other_tax_amount>5.67</other_tax_amount>
        <merchant_vat_no>merchant_vat_no</merchant_vat_no>
        <customer_vat_no>cus_vat_no</customer_vat_no>
        <destination_zip>M1R 1W4</destination_zip>
        <ship_from_zip>M6K 2H9</ship_from_zip>
      </addendum1>
      <addendum2>

```

```
<item_description>item_description</item_description>
<product_code>product_code</product_code>
<quantity>12</quantity>
<unit_cost>1.2345</unit_cost>
<ext_amount>2.89</ext_amount>
<uom>EA</uom>
<item_discount_amount>6.78</item_discount_amount>
<item_tax_amount>4.32</item_tax_amount>
<item_other_tax_amount>1.94</item_other_tax_amount>
<item_other_tax_type>QST</item_other_tax_type>
</addendum2>
</batch_l23ind_refund>
<batch_l23forcepost>
  <order_id>NQA_20140428_3</order_id>
  <cust_id>custid123</cust_id>
  <amount>1.00</amount>
  <pan>4242-424254545454</pan>
  <expdate>1512</expdate>
  <auth_code>964734</auth_code>
  <crypt_type>7</crypt_type>
  <addendum1>
    <customer_code>customer_code</customer_code>
    <discount_amount>1.12</discount_amount>
    <freight_amount>2.23</freight_amount>
    <duty_amount>3.34</duty_amount>
    <national_tax_amount>4.56</national_tax_amount>
    <other_tax_amount>5.67</other_tax_amount>
    <merchant_vat_no>merchant_vat_no</merchant_vat_no>
    <customer_vat_no>cus_vat_no</customer_vat_no>
    <destination_zip>M1R 1W4</destination_zip>
    <ship_from_zip>M6K 2H9</ship_from_zip>
  </addendum1>
  <addendum2>
    <item_description>item_description</item_description>
    <product_code>product_code</product_code>
    <quantity>12</quantity>
    <unit_cost>1.2345</unit_cost>
    <ext_amount>2.89</ext_amount>
    <uom>EA</uom>
    <item_discount_amount>6.78</item_discount_amount>
    <item_tax_amount>4.32</item_tax_amount>
    <item_other_tax_amount>1.94</item_other_tax_amount>
    <item_other_tax_type>QST</item_other_tax_type>
  </addendum2>
</batch_l23forcepost>
</request>
</l23_batch>
```

17. Appendix G: XML Response XML

Below is the XML response of the Appendix F: XML Request Example.



The response below has been re-formatted with spacing and newlines for documentation purpose. In the actual response, the whole response is a single line, so you should be parsing for XML tags rather than newline.

```
<123_batch>
  <response>
    <orig_request>
      <request>
        <store_id>moneris</store_id>
        <api_token>hurgle</api_token>
        <batch_123purchase>
          <order_id>NQA_20140428_0</order_id>
          <cust_id>custid123</cust_id>
          <amount>1.00</amount>
          <pan>4242-****5454</pan>
          <expdate>1512</expdate>
          <crypt_type>7</crypt_type>
          <addendum1>
            <customer_code>customer_code</customer_code>
            <discount_amount>1.12</discount_amount>
            <freight_amount>2.23</freight_amount>
            <duty_amount>3.34</duty_amount>
            <national_tax_amount>4.56</national_tax_amount>
            <other_tax_amount>5.67</other_tax_amount>
            <merchant_vat_no>merchant_vat_no</merchant_vat_no>
            <customer_vat_no>cus_vat_no</customer_vat_no>
            <destination_zip>M1R 1W4</destination_zip>
            <ship_from_zip>M6K 2H9</ship_from_zip>
          </addendum1>
          <addendum2>
            <item_description>item_description</item_description>
            <product_code>product_code</product_code>
            <quantity>12</quantity>
            <unit_cost>1.2345</unit_cost>
            <ext_amount>2.89</ext_amount>
            <uom>EA</uom>
            <item_discount_amount>6.78</item_discount_amount>
            <item_tax_amount>4.32</item_tax_amount>
            <item_other_tax_amount>1.94</item_other_tax_amount>
            <item_other_tax_type>QST</item_other_tax_type>
          </addendum2>
        </batch_123purchase>
      </request>
    </orig_request>
    <receipt>
      <ReceiptId>NQA_20140428_0</ReceiptId>
      <ReferenceNum>660117310013420270</ReferenceNum>
      <ResponseCode>005</ResponseCode>
      <ISO>01</ISO>
      <AuthCode>996941</AuthCode>
      <TransTime>16:14:44</TransTime>
      <TransDate>2014-04-28</TransDate>
      <TransType>02</TransType>
      <Complete>true</Complete>
      <Message>APPROVED *
n stored in database.</Message>
```

=: Level 2/3 message have bee


```

    <TransAmount>1.00</TransAmount>
    <CardType>V</CardType>
    <TransID>97634-1_9</TransID>
    <TimedOut>>false</TimedOut>
    <BankTotals>>null</BankTotals>
    <Ticket>>null</Ticket>
    <CorporateCard>>true</CorporateCard>
    <MessageId>1A4118584849702</MessageId>
  </receipt>
</response>
<response>
  <orig_request>
    <request>
      <store_id>moneris</store_id>
      <api_token>hurgle</api_token>
      <batch_l23ind_refund>
        <order_id>NQA_20140428_1</order_id>
        <cust_id>custid123</cust_id>
        <amount>1.00</amount>
        <pan>4242-****5454</pan>
        <expdate>1512</expdate>
        <crypt_type>7</crypt_type>
        <addendum1>
          <customer_code>customer_code</customer_code>
          <discount_amount>1.12</discount_amount>
          <freight_amount>2.23</freight_amount>
          <duty_amount>3.34</duty_amount>
          <national_tax_amount>4.56</national_tax_amount>
          <other_tax_amount>5.67</other_tax_amount>
          <merchant_vat_no>merchant_vat_no</merchant_vat_no>
          <customer_vat_no>cus_vat_no</customer_vat_no>
          <destination_zip>M1R 1W4</destination_zip>
          <ship_from_zip>M6K 2H9</ship_from_zip>
        </addendum1>
        <addendum2>
          <item_description>item_description</item_description>
          <product_code>product_code</product_code>
          <quantity>12</quantity><unit_cost>1.2345</unit_cost>
          <ext_amount>2.89</ext_amount>
          <uom>EA</uom>
          <item_discount_amount>6.78</item_discount_amount>
          <item_tax_amount>4.32</item_tax_amount>
          <item_other_tax_amount>1.94</item_other_tax_amount>
          <item_other_tax_type>QST</item_other_tax_type>
        </addendum2>
      </batch_l23ind_refund>
    </request>
  </orig_request>
</receipt>
  <ReceiptId>NQA_20140428_1</ReceiptId>
  <ReferenceNum>660117310013420280</ReferenceNum>
  <ResponseCode>005</ResponseCode>
  <ISO>01</ISO>
  <AuthCode>010237</AuthCode>
  <TransTime>16:14:45</TransTime>
  <TransDate>2014-04-28</TransDate>
  <TransType>04</TransType>
  <Complete>>true</Complete>
  <Message>APPROVED *
n stored in database.</Message>
  <TransAmount>1.00</TransAmount>
  <CardType>V</CardType>
  <TransID>97635-0_9</TransID>
  <TimedOut>>false</TimedOut>

```

=: Level 2/3 message have bee

```

    <BankTotals>null</BankTotals>
    <Ticket>null</Ticket>
    <CorporateCard>true</CorporateCard>
    <MessageId>1A4118584859712</MessageId>
  </receipt>
</response>
<response>
  <orig_request>
    <request>
      <store_id>moneris</store_id>
      <api_token>hurgle</api_token>
      <batch_123forcepost>
        <order_id>NQA_20140428_3</order_id>
        <cust_id>custid123</cust_id>
        <amount>1.00</amount>
        <pan>4242-****5454</pan>
        <expdate>1512</expdate>
        <auth_code>964734</auth_code>
        <crypt_type>7</crypt_type>
        <addendum1>
          <customer_code>customer_code</customer_code>
          <discount_amount>1.12</discount_amount>
          <freight_amount>2.23</freight_amount>
          <duty_amount>3.34</duty_amount>
          <national_tax_amount>4.56</national_tax_amount>
          <other_tax_amount>5.67</other_tax_amount>
          <merchant_vat_no>merchant_vat_no</merchant_vat_no>
          <customer_vat_no>cus_vat_no</customer_vat_no>
          <destination_zip>M1R 1W4</destination_zip>
          <ship_from_zip>M6K 2H9</ship_from_zip>
        </addendum1>
        <addendum2>
          <item_description>item_description</item_description>
          <product_code>product_code</product_code>
          <quantity>12</quantity>
          <unit_cost>1.2345</unit_cost>
          <ext_amount>2.89</ext_amount>
          <uom>EA</uom>
          <item_discount_amount>6.78</item_discount_amount>
          <item_tax_amount>4.32</item_tax_amount>
          <item_other_tax_amount>1.94</item_other_tax_amount>
          <item_other_tax_type>QST</item_other_tax_type>
        </addendum2>
      </batch_123forcepost>
    </request>
  </orig_request>
</receipt>
  <ReceiptId>NQA_20140428_3</ReceiptId>
  <ReferenceNum>660117310013420290</ReferenceNum>
  <ResponseCode>005</ResponseCode>
  <ISO>01</ISO>
  <AuthCode>964734</AuthCode>
  <TransTime>16:14:46</TransTime>
  <TransDate>2014-04-28</TransDate>
  <TransType>02</TransType>
  <Complete>true</Complete>
  <Message>APPROVED *
  <TransAmount>1.00</TransAmount>
  <CardType>V</CardType>
  <TransID>97636-0_9</TransID>
  <TimedOut>false</TimedOut>
  <BankTotals>null</BankTotals>
  <Ticket>null</Ticket>

```

n stored in

database.</Message>

=: Level 2/3 message have bee

```
<CorporateCard>true</CorporateCard>
  <MessageId>1A4118584869720</MessageId>
</receipt>
</response>
</123_batch>
```

eSELECTplus™

Copyright Notice

Copyright © 2014 Moneris Solutions, 3300 Bloor Street West, Toronto, Ontario, M8X 2X2

All Rights Reserved. This manual shall not wholly or in part, in any form or by any means, electronic, mechanical, including photocopying, be reproduced or transmitted without the authorized, written consent of Moneris Solutions.

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

Moneris and the Moneris Solutions logo are registered trademarks of Moneris Solutions Corporation.

Any software, hardware and or technology products named in this document are claimed as trademarks or registered trademarks of their respective companies.

Printed in Canada.

10 9 8 7 6 5 4 3 2 1