PayDollar PayGate

Integration Guide version 3.7

(Leave Blank Intentionally)

Copyright Information

AsiaPay (HK) Limited
Unit 1701-2, 17/F, K. Wah Centre
191 Java Road
Hong Kong.
Telephone ☹852) 2538 8278
Fax: (852) 2545 3898
Web site: http://www.asiapay.com

Confidentiality and Usage

The information contained in this document (and any attachments) is confidential information provided by AsiaPay (HK) Limited. This document is intended only for use by merchants approved by AsiaPay. Any copying, distribution or dissemination of this document by any other parties is prohibited.

Revision History

Revision	Date	Revision Description
0.0	July 11,2001	First Draft issue
1.0	Apr 28, 2003	First issue (All in One Copy)
1.1	Oct 7, 2003	Add VBV Parameters for Server Side Integration
1.2	Nov 1, 2003	Modify VBV Parameters for Server Side Integration
1.3	Nov 6, 2003	Deleted Data Feed Retry Message
1.4	Nov 13, 2003	Add AuthId to Data Feed Output
1.5	Feb 20, 2004	Add JCB Card Type and Japanese Lang Parameter
1.6	Mar 9, 2004	All-in-One Version
1.7	May 6, 2004	Add JPY and TWD Currency Code
1.8	May 13, 2004	Add data feed flow comment
1.9	Aug. 20, 2004	Add AMEX and Security Code parameter
2.0	Sept 14, 2004	Add Payment Flow, 3D Transaction, PayPal Transaction and New Currencies
2.1	Oct 8, 2004	Modify the pop-up flow to be inline flow of 3D Transaction
2.2	Dec 6, 2004	Add errorUrl parameter to Direct Client Side Connection
2.3	Sep 15, 2005	Add ECI and Payer Authentication Status to datafeed section
2.5	Jan 25, 2006	Add data feed port requirement
2.6	Mar 30, 2006	Add new parameters to datafeed : - sourceIp - ipCountry
2.7	Jun 15, 2006	Add sample data feed page in ASP.
2.8	Jun 20, 2006	Modified Payment Method to support CUP.
2.9	Nov 27, 2006	Add Transaction Handling and FAQ Section
3.0	Aug 30, 2007	Add Alipay payment method
3.1	Jan 24, 2008	Language update – Supporting Thai
3.11	June, 25, 2008	Revise and updated content
3.2	Mar 25, 2009	Add China UnionPay (CHINAPAY) payment method
3.2.1	Apr 29, 2009	Add new parameter “payMethod” to Data Feed Output
3.2.2	May 14, 2009	Add APPENDIX
3.3	Nov 5, 2009	Add Tenpay and 99bill payment method
3.4	Dec 18, 2009	Add new optional parameter for connect to payment page. Origin Country and Destination Country
3.5	Jun 1, 2010	Add new parameter “cardIssuingCountry” to Data Feed Output
3.6	Jun 29,2010	Add new function “Multi-Currency Processing” and merge “Server Side Direct Connection” Spec into one spec. Move “Data Feed Handling” into a new section. Add mps fields in datafeed and merchant api.
3.7	Jul 27, 2010	Add Secure Hash Function. Add new optional parameter “amount” in Merchant API RequestRefund action

Table of Contents

1 Overview

1.1 Introduction

PayDollar PayGate is a powerful web-based online payment services platform, which provides secure, multi-channel, multi-lingual and multi-currency payment services. It is used by many renowned companies and organizations in the region.

This technical specification document prescribes the constituent parts of specification for integration of an e-commerce web site with PayDollar e-commerce service, the on-line payment service by AsiaPay (HK) Limited, by subscribed merchants of the service. This document has been created to ensure that all technical specifications contain sufficient information to enable a merchant to design and modify the codes of an existing on-line shopping architecture or software to cater for the payment –enabling service. It also provides a checklist to enable the reviewers of specifications to conduct tests on the functionalities of the integration.

PayDollar PayGate facilitates merchant to connect to our network with great flexibility. Merchant can choose one of the following integration methods, which will be described in detail in the document.

- Client Post through Browser (e.g. Shopping Cart)
- Direct Client Side Connection
- Server Side Direct Connection

Moreover, a list of merchant API functions will be also described in detail in the later section.

2 Connection method

2.1 Client Post Through Browser

It is the most popular connection method among merchants. The advantage of this connection method is simple and speedy. On the other hand, payment transaction flow is ready to use. Merchant can kick off the web site on-the-fly with just a small scale integration.

Scope and Compatibility

This connection is designed for merchants who have *Online Shopping Cart System*. The routine is HTML-based with Javascript and should be widely applicable to on-line shopping cart software and architecture, whose technical specifications and varieties are beyond the scope of this document. Compatibility with shopping cart software is yet to be exhaustively given and would not be included in the scope of this document.

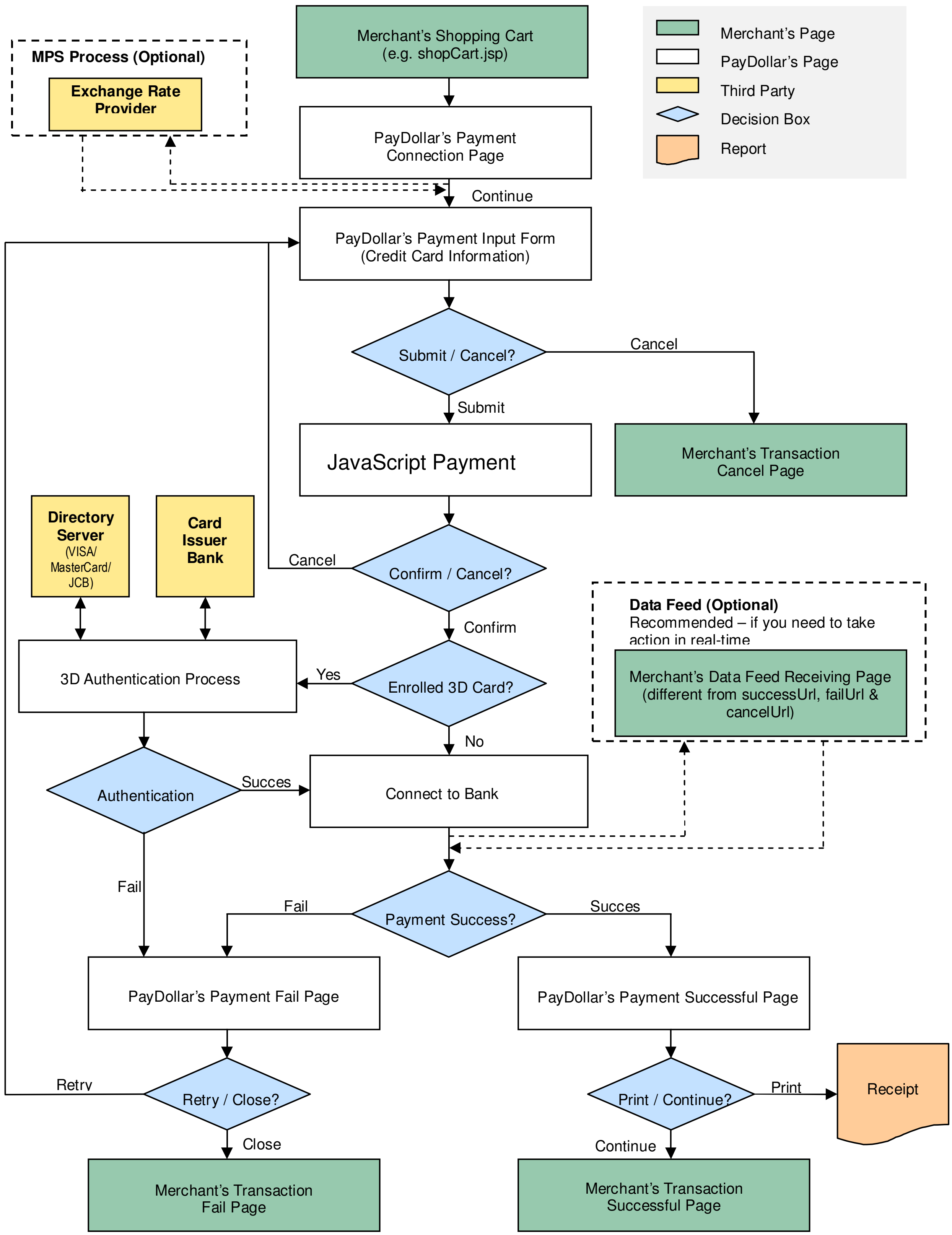
The compatible version of the software code is as follow:

Software Code	Version
HTML	4.0
Javascript	1.3

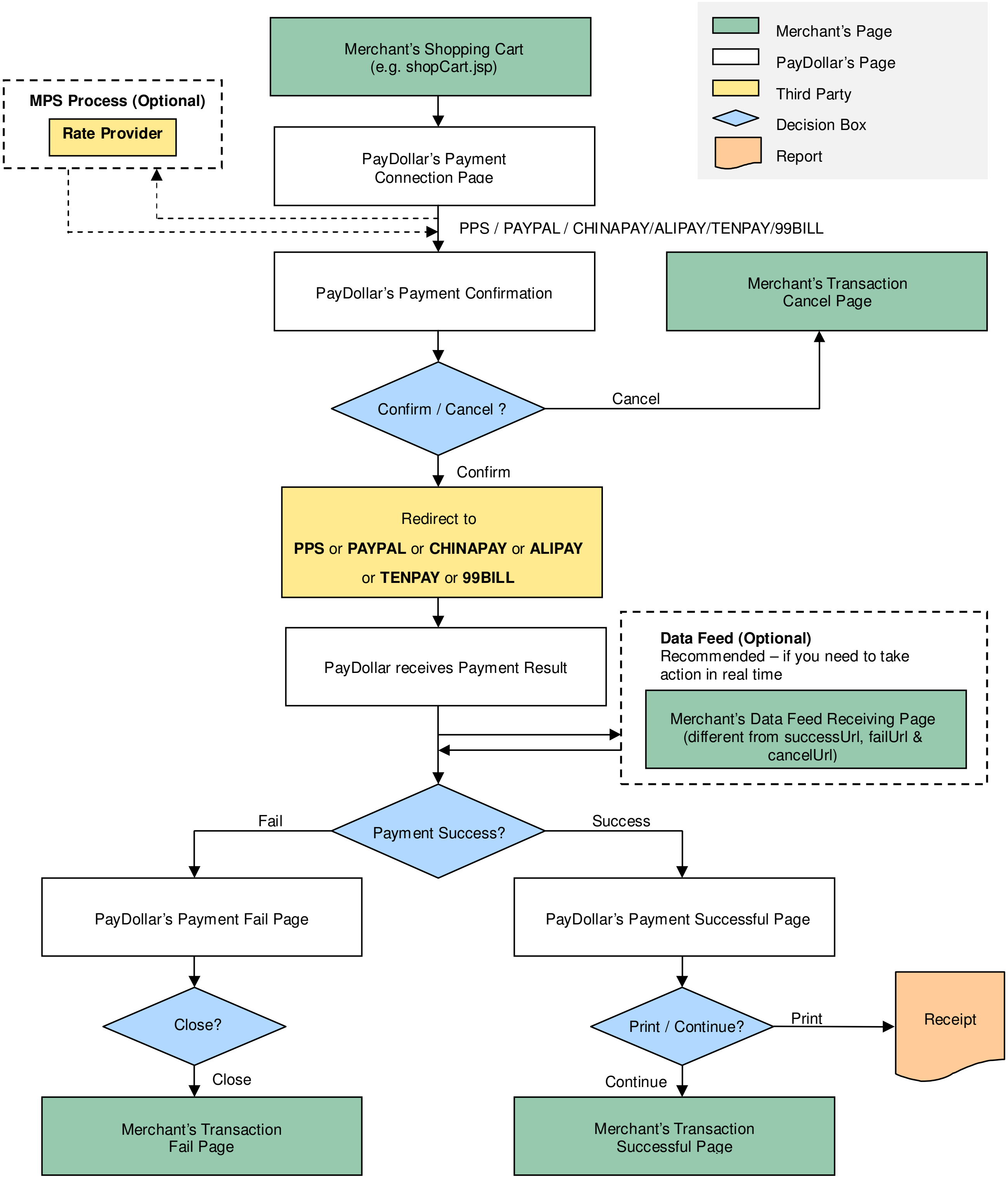
The version compatibility of the software code with popular browser software is as follow:

Browser	Version
Microsoft Internet Explorer	5.0 or above
Netscape Netvigator (English version)	4.7 or above

Credit Card Payment Flow


MPS Process (Optional)
Exchange Rate Provider
Merchant's Shopping Cart (e.g. shopCart.jsp)
Merchant's Page
PayDollar's Page
Third Party
Decision Box
Report
PayDollar's Payment Connection Page
Continue
PayDollar's Payment Input Form (Credit Card Information)
Submit / Cancel?
Cancel
Submit
JavaScript Payment
Merchant's Transaction Cancel Page
Directory Server (VISA/ MasterCard/ JCB)
Card Issuer Bank
Cancel
Confirm / Cancel?
Confirm
Data Feed (Optional)
Recommended – if you need to take action in real-time
Merchant's Data Feed Receiving Page (different from successUrl, failUrl & cancelUrl)
3D Authentication Process
Yes
Enrolled 3D Card?
No
Authentication
Succes
Connect to Bank
Fail
Fail
Payment Success?
Succes
PayDollar's Payment Fail Page
PayDollar's Payment Successful Page
Retry
Retry / Close?
Close
Print / Continue?
Print
Receipt
Continue
Merchant's Transaction Fail Page
Merchant's Transaction Successful Page

99BILL, ALIPAY, CHINAPAY, PAYPAL, PPS, TENPAY Payment Flow


Merchant's Page
PayDollar's Page
Third Party
Decision Box
Report
Merchant's Shopping Cart (e.g. shopCart.jsp)
MPS Process (Optional)
Rate Provider
PayDollar's Payment Connection Page
PPS / PAYPAL / CHINAPAY/ALIPAY/TENPAY/99BILL
PayDollar's Payment Confirmation
Merchant's Transaction Cancel Page
Confirm / Cancel ?
Cancel
Confirm
Redirect to
PPS or PAYPAL or CHINAPAY or ALIPAY or TENPAY or 99BILL
PayDollar receives Payment Result
Data Feed (Optional)
Recommended – if you need to take action in real time
Merchant's Data Feed Receiving Page (different from successUrl, failUrl & cancelUrl)
Payment Success?
Fail
Success
PayDollar's Payment Fail Page
PayDollar's Payment Successful Page
Close?
Close
Print / Continue?
Print
Receipt
Continue
Merchant's Transaction Fail Page
Merchant's Transaction Successful Page

It is notable that the software codes of the payment routine, whose example is as given, should be embedded into the integration page, as in above, which should be able to generate the sum of purchase from the previous shopping practice of the user. The subsequent parts of the flow would be directed to system architecture in which the payment details are to be submitted by the user, and handled for settlement and clearance.

Definition of Parameters in the Integration Page

The following are the parameters for integration. PayDollar PayGate is case sensitive. Make sure the typeface is correct. When a transaction is finished, the system will return customer a payment message. Merchant can create static HTML pages to display the message. If merchant’s web site supports data feed, the system can return payment message as shown in the following table.

Parameters	Data Type	Descriptions
Required Parameter (with UTF-8 Encoding) for connect to our payment page		
orderRef	Text (35)	Merchant‘s Order Reference Number
mpsMode	Text(3)	The Multi-Currency Processing Service (MPS) Mode: “NIL” or not provide – Disable MPS (merchant not using MPS) “SCP” – Enable MPS with ‘Simple Currency Conversion’ “DCC” – Enable MPS with ‘Dynamic Currency Conversion’ “MCP” – Enable MPS with ‘Multi Currency Pricing’ For merchant who applied MPS function
currCode	Text (3)	The currency of the payment: “344” – HKD “840” – USD “702” – SGD “156” – CNY (RMB) “392” – JPY “901” – TWD “036” – AUD “978” – EUR “826” – GBP “124” – CAD Remark: For MPS mode set with SCP, the currCode should be in the foreign currency.

amount	Number (12,2)	The total amount your want to charge the customer for the provided currency Remark: For MPS mode set with SCP, the amount should be in the foreign currency.
lang	Text (1)	The language of the payment page i.e. “C” – Traditional Chinese “E” – English “X” – Simplified Chinese “K” – Korean “J” – Japanese “T” – Thai
cancelUrl	Text (300)	A Web page address you want us to redirect upon the transaction being cancelled by your customer **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**
failUrl	Text (300)	A Web page address you want us to redirect upon the transaction being rejected by us. **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**
successUrl	Text (300)	A Web page address you want us to redirect upon the transaction being accepted by us **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**
merchantId	Number	The merchant ID we provide to you
payType	Text(1); (“N”,”H”)	The payment type: ”N” – Normal Payment (Sales) ”H” – Hold Payment (Authorize only) Hold Payment is not available for 99BILL, ALIPAY, CHINAPAY, PAYPAL, PPS, TENPAY
payMethod	Text; (“ALL”,”PPS”,”CC”, “PAYPAL”, ”CHINAPAY”, “ALIPAY”, “TENPAY”, “99BILL)	The payment method: “ALL” – All the available payment method “CC” – Credit Card Payment “PPS” – PayDollar PPS Payment “PAYPAL” – PayPal By PayDollar Payment “CHINAPAY” – China UnionPay By PayDollar Payment “ALIPAY” – ALIPAY By PayDollar Payment

		"TENPAY" – TENPAY BY PayDollar Payment "99BILL" – 99BILL BY PayDollar Payment
Optional Parameter for connect to our payment page		
remark	Text (200)	A remark field for you to store additional data that will not show on the transaction web page
redirect	Number	Number of seconds auto-redirection to merchant's site takes place at PayDollar's Payment Success / Fail page
oriCountry	Number(3)	Origin Country Code Example: 344 – "HK" 840 – "US"
destCountry	Number(3)	Destination Country Code Example: 344 – "HK" 840 – "US"
secureHash	Text (40)	Secure hash is used to authenticate the integrity of the transaction information and the identity of the merchant. It is calculated by hashing the combination of various transaction parameters and the Secure Hash Secret. *Applies to merchants who registered this function only. For more information, please refer to section 4.
Redirect URL (successUrl, failUrl and cancelUrl) Output		
Ref	Text	Merchant's Order Reference Number **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**

Example of Client Post Method (Source Code)

The following is an example of integration of shopping cart routine with the payment routine of PayDollar PayGate in HTML. It is noteworthy that the portion in bold typeface as follows is mandatory for successful integration.

In the following sample form, hidden fields are used to hold the values:

```
...
<form name="payFormCcard" method="post" action="
        https://test.paydollar.com/b2c2/eng/payment/payForm.jsp">
<input type="hidden" name="merchantId" value="1">
<input type="hidden" name="amount" value="3000.0" >
```

```
<input type="hidden" name="orderRef" value="000000000014">
<input type="hidden" name="currCode" value="344" >
<input type="hidden" name="mpsMode" value="NIL" >
<input type="hidden" name="successUrl"
        value="http://www.yourdomain.com/Success.html">
<input type="hidden" name="failUrl" value="http://www.yourdomain.com/Fail.html">
<input type="hidden" name="cancelUrl" value="http://www.yourdomain.com/Cancel.html">
<input type="hidden" name="payType" value="N">
<input type="hidden" name="lang" value="E">
<input type="hidden" name="payMethod" value="CC">
<input type=”hidden” name=”secureHash” value=” 44f3760c201d3688440f62497736bfa2aadd1bc0”>
<input type="submit" name="submit">
</form>
...
```

Kick Off

After the integration has been completed, it is ready to launch your e-commerce web to serve your customers. Please copy the following **TESTING URL** for client post method:

`https://test.paydollar.com/b2cDemo/eng/payment/payForm.jsp`

Please copy the following **PRODUCTION URL** for client post method:

`https://www.paydollar.com/b2c2/eng/payment/payForm.jsp`

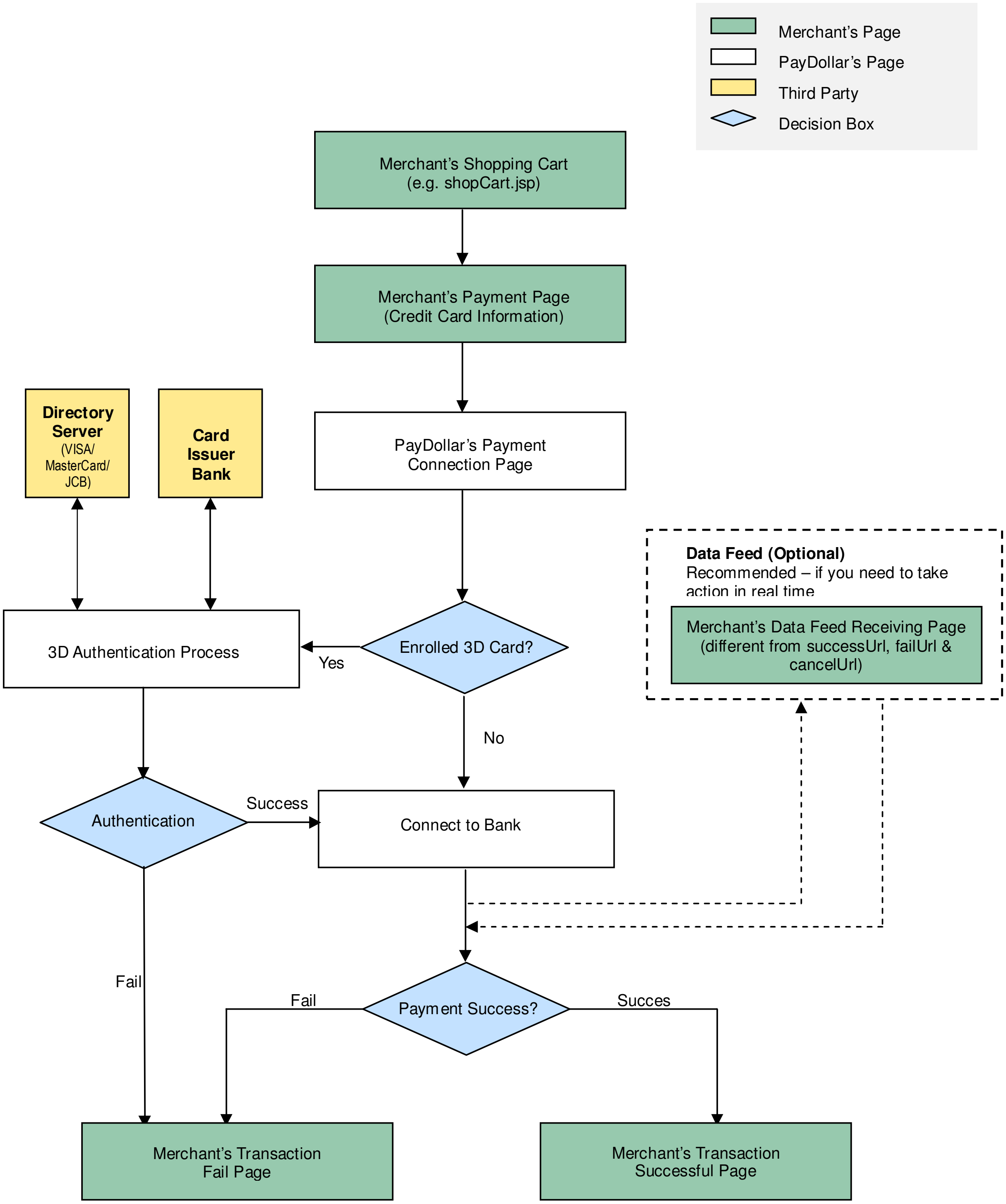
2.2 Direct Client Side Connection

This method is used for the merchant if they want to capture the credit card information from their web page instead of using our standard payment page. This connection method only apply to credit card transaction that 99BILL, ALIPAY, CHINAPAY, PayPal, PPS and TENPAY transaction is not allow for using this method. The requirement of using this method is to install a SSL Certificate to your domain in order to protect your customers' credit card information.

Moreover, if the credit card used by the customer is a registered 3D card, the customer will be asked for providing a personal password to verify the payer identity. 3D Secure is a credit card authorization program implemented by VISA with brand named "Verified By VISA", MasterCard with brand named "MasterCard SecureCode" and JCB with brand named "J/Secure" to reduce fraudulent purchases by verifying purchaser identity during online transactions. PayDollar will assist to carry out this process and the customer will observe the 3D processing pages by our PayDollar shown as the later section.

As the 3D protocol is standardized for all brand types, including Verified By VISA, MasterCard SecureCode and JCB J/Secure. In this document, we use the case of Verified By VISA as an example to show the flow in detail.

Payment Flow

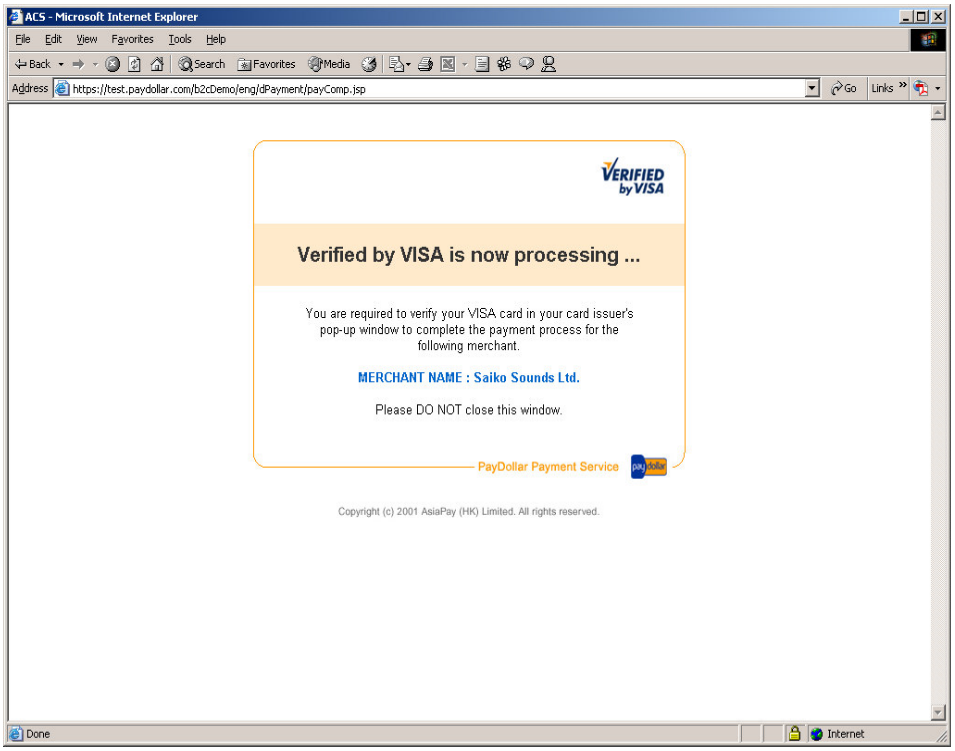

Merchant's Page
PayDollar's Page
Third Party
Decision Box
Merchant's Shopping Cart (e.g. shopCart.jsp)
Merchant's Payment Page (Credit Card Information)
Directory Server (VISA/ MasterCard/ JCB)
Card Issuer Bank
PayDollar's Payment Connection Page
Data Feed (Optional)
Recommended – if you need to take action in real time
Merchant's Data Feed Receiving Page (different from successUrl, failUrl & cancelUrl)
3D Authentication Process
Yes
Enrolled 3D Card?
No
Authentication
Success
Connect to Bank
Fail
Fail
Payment Success?
Succes
Merchant's Transaction Fail Page
Merchant's Transaction Successful Page

Non-3D transaction

Your client's browser will be redirected from your site to our payment page and then we will redirect the page to your successful/fail page upon completed the transaction.

3D transaction

As 3D Authentication require your customers to enter the password of their cards, your clients' browser will be redirected to a 3D notification web page in order to notify your customers that they need to complete the 3D Authentication by entering the password in the pop-up window. Below are some sample pages for the case of Verified By VISA.


ACS - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go Links
VERIFIED by VISA
PayDollar Payment Service
Done
Internet

https://test.paydollar.com/b2cDemo/eng/dPayment/payComp.jsp

Verified by VISA is now processing ...

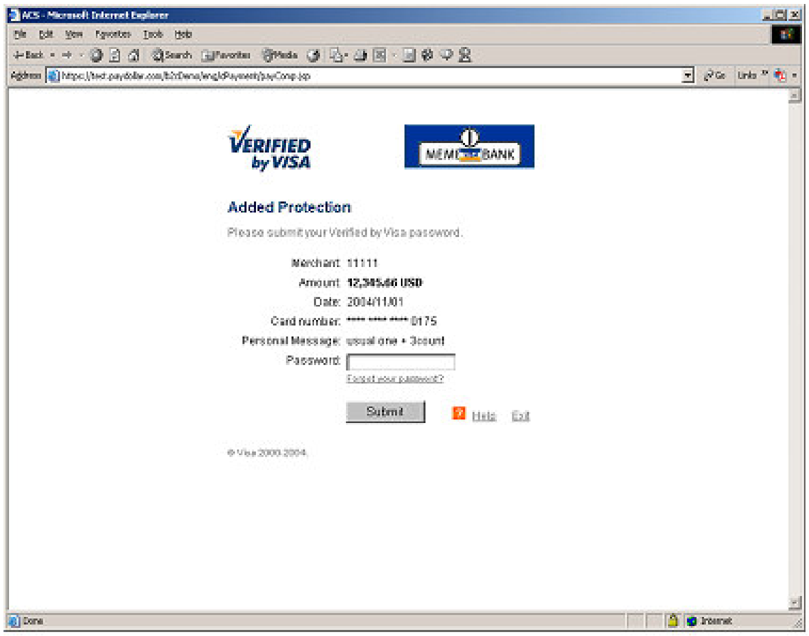
You are required to verify your VISA card in your card issuer's pop-up window to complete the payment process for the following merchant.

MERCHANT NAME : Saiko Sounds Ltd.

Please DO NOT close this window.

Copyright (c) 2001 AsiaPay (HK) Limited. All rights reserved.

Figure 1.1 Sample notification page


ACS - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go Links
VERIFIED by VISA
MEMBER BANK
Help Exit
Done
Internet

Added Protection

Please submit your Verified by Visa password.

Merchant: 11111
Amount: **12,345.66 USD**
Date: 2004/11/01
Card number: **** **** **** 0175
Personal Message: usual one + 3count
Password:
Forgot your password?

Submit

© Visa 2000-2004.

Figure 1.2 Sample issuing bank verification page

After the authentication process, the authentication result will forward to our system and the transaction process will be further continued by our acquiring bank according to the authentication result.

Integration Procedures

To connect to our system, you need to post the required parameters to our payment page URL and then get back the result by using data feed.

Definition of Parameters in the Integration Page

In the targeted page of integration, in which **sum of purchase** has been generated, the following fields (hidden or text) should be added:

Parameters	Data Type	Descriptions
Required Parameter (with UTF-8 Encoding) for connect to our payment page		
orderRef	Text (35)	Merchant's Order Reference Number
amount	Number (12,2)	The total amount your want to charge the customer (up to 2 decimal place)
currCode	Text (3)	The currency of the payment: "344" - HKD "840" – USD "702" – SGD "156" – CNY (RMB) "392" – JPY "901" – TWD "036" – AUD "978" – EUR "826" – GBP "124" – CAD
lang	Text (1) ("C","E","X","K", "J", "T")	The language of the payment page : "C" - Traditional Chinese "E" - English "X" - Simplified Chinese "K" – Korean "J" – Japanese "T" – Thai
merchantId	Number	The merchant ID we provide to you
pMethod	Text	The payment card type (e.g. "VISA", "Master", "Diners", "JCB", "AMEX")
epMonth	Number(2)	Credit card expiry month
epYear	Number(4)	Credit card expiry year
cardNo	Text (16)	Credit card number

securityCode	Text (4)	Credit Card Verification Code - VISA: CVV2 (3-digit) - MasterCard: CVC2 (3-digit) - American Express: 4DBC (4-digit)
cardHolder	Text (20)	Credit card holder name
failUrl	Text (300)	A Web page address you want us to redirect upon the transaction being rejected by us **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**
successUrl	Text (300)	A Web page address you want us to redirect upon the transaction being accepted by us **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**
errorUrl	Text (300)	A Web page address you want us to redirect when unexpected error occur (e.g. parameter incorrect) **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**
payType	Text (1) (“N”,”H”)	The payment type: “N” – Normal Payment (Sales) “H” – Hold Payment (Authorize only)
Optional Parameter for connect to our payment page		
remark	Text	An additional remark field that will appear in the confirmation email and transaction detail report to help you to refer the order
oriCountry	Number(3)	Origin Country Code Example: 344 – “HK” 840 – “US”
destCountry	Number(3)	Destination Country Code Example: 344 – “HK” 840 – “US”
secureHash	Text (40)	Secure hash is used to authenticate the integrity of the transaction information and the identity of the merchant. It is calculated by hashing the combination of various transaction parameters and the Secure Hash Secret. *Applies to merchants who registered this function only. For more information, please refer to section 4.
Redirect URL (successUrl, failUrl and errorUrl) Output		
Ref	Text	Merchant’s Order Reference Number **(For display purpose only. DO NOT use this URL to update your system. Please use DataFeed for this purpose.)**

Example of connecting to our gateway (Direct Client Side Connection)

As different type of programming language have different syntax. Therefore, the sample code below, is written in HTML code, the requirement is to form post all the required parameters to our secure API, hilighted in yellow.

Sample code:

```
...
<form name="payForm" method="post"
action="https://test.paydollar.com/b2cDemo/eng/dPayment/payComp.jsp">
<input type="hidden" name="merchantId" value="1">
<input type="hidden" name="amount" value="3000.0" >
<input type="hidden" name="orderRef" value="000000000006">
<input type="hidden" name="currCode" value="344" >
<input type="hidden" name="pMethod" value="VISA" >
<input type="hidden" name="cardNo" value="4918914107195005" >
<input type="hidden" name="securityCode" value="123" >
<input type="hidden" name="cardHolder" value="Testing" >
<input type="hidden" name="epMonth" value="07” >
<input type="hidden" name="epYear" value="2009" >
<input type="hidden" name="payType” value="N" >
<input type="hidden" name="successUrl"
value="http://www.yourwebsite.com/pSuccess.jsp">
<input type="hidden" name="failUrl" value="http://www.yourwebsite.com/pFail.jsp">
<input type="hidden" name="errorUrl" value="http://www.yourwebsite.com/pError.jsp">
<input type="hidden" name="lang" VALUE="E">
<input type=”hidden” name=”secureHash” value=” 44f3760c201d3688440f62497736bfa2aadd1bc0”>
<input type="submit" value="Pay Now">
</form>
...
```

** All the source code in this document are the property of AsiaPay (HK) Limited. Any use, modification and adaptation to the code should be reported to and approved by AsiaPay (HK) Limited. AsiaPay (HK) Limited do not have any liability in any lose to the party using the source code.*

Kick Off

After the integration has been completed, it is ready to launch your e-commerce web to serve your customers. Please copy the following **TESTING URL** for client post method:

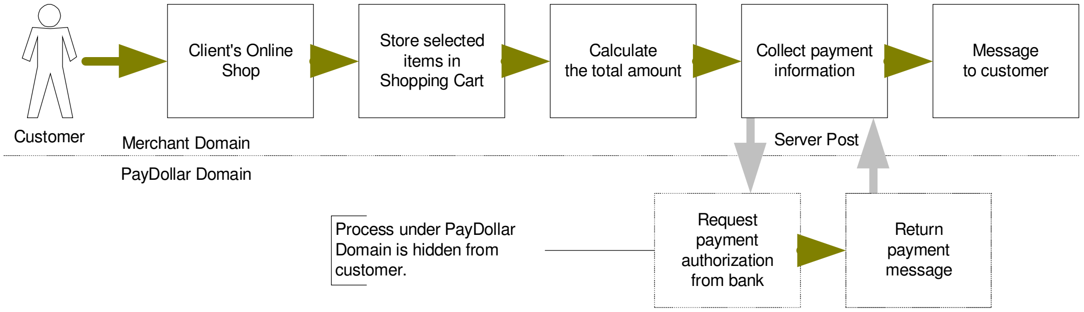
`https://test.paydollar.com/b2cDemo/eng/dPayment/payComp.jsp`

Please copy the following **PRODUCTION URL** for client post method:

`https://www.paydollar.com/b2c2/eng/dPayment/payComp.jsp`

2.3 Server Side Direct Connection

This connection method is for merchant to request payment authorization from bank directly through PayDollar PayGate system. In this connection, merchants need to build their own payment information collection page to collect payment information, such as credit card number, expire data, holder's name and etc. Then, payment information has to be sent to a defined URL provided by the acquiring bank. Customer of the merchant, therefore, will not see any bank's payment page.


Customer
Client's Online Shop
Store selected items in Shopping Cart
Calculate the total amount
Collect payment information
Message to customer
Merchant Domain
Server Post
PayDollar Domain
Process under PayDollar Domain is hidden from customer.
Request payment authorization from bank
Return payment message

For merchant who chooses this method of connection, 128-bit SSL cert must be installed for data encryption. The system does not accept non-encrypted data.

PayDollar use Extended Validation (EV) SSL Certificate. To ensure your system function properly, please check your certificate store can recognize VeriSign intermediate CA certificate - Secure Site Pro/Managed PKI for SSL Premium with EV Certificates. If not, you are required to install the VeriSign intermediate CA certificate in your certificate store.

Please download the primary and secondary VeriSign EV SSL Intermediate CA certificates from the following link then import the 2 certificates into the keystore of your environment.

http://www.verisign.com/support/verisign-intermediate-ca/extended-validation-pro/index.html

(Please be reminded that you should choose the option "Issued After May 17th, 2009")

Definition of Parameters in the Integration Page

The following are the parameters for integration. PayDollar PayGate is case sensitive. Make sure the typeface is correct. When a transaction is finish, the system will return customer a payment message on the page created by merchant.

Parameters	Data Type	Descriptions
Required Parameter (with UTF-8 Encoding) for connect to our payment interface		
orderRef	Text (35)	Merchant's Order Reference Number
amount	Number (12,2)	Total amount your want to charge the customer [Up to 2 decimal place]
currCode	Number ("344","840","702","156", "392", "901")	The currency of the payment: "344" - HKD "840" – USD "702" – SGD "156" – CNY (RMB) "392" – JPY "901" – TWD "036" – AUD "978" – EUR "826" – GBP "124" – CAD
lang	Text (1) ("C","E","X", "K", "J")	The language of the payment page : "C" - Traditional Chinese "E" - English "X" - Simplified Chinese "K" – Korean "J" – Japanese
merchantId	Number	The merchant ID we provide to you
pMethod	Text ("VISA","Master","Diners","JCB", " AMEX")	The payment card type
epMonth	Number(2)	Credit card expiry month
epYear	Number(4)	Credit card expiry year
cardNo	Text (16)	Credit card number
cardHolder	Text (20)	Credit card holder name
securityCode	Text (4)	Credit Card Verification Code - VISA: CVV2 (3-digit)

		- MasterCard: CVC2 (3-digit) - American Express: 4DBC (4-digit)
payType	Text (1) ("N","H")	The payment type: "N" – Normal Payment (Sales) "H" – Hold Payment (Authorize only)
Optional Parameter for connect to our payment interface		
remark	Text	An additional remark field that will appear in the confirmation email and transaction detail report to help you to refer the order
secureHash	Text (40)	Secure hash is used to authenticate the integrity of the transaction information and the identity of the merchant. It is calculated by hashing the combination of various transaction parameters and the Secure Hash Secret. *Applies to merchants who registered this function only. For more information, please refer to section 4.
Parameter For 3D Transaction (Need to install MPI Server Software at Merchants' site)		
vbvTransaction	Text ("T","F")	3D Transaction (i.e. Verified By VISA, MasterCard SecureCode, J/Secure) "T" – True: The transaction has been gone through the VE/PA process. "F" – False: Non-3D transaction without go through the VE/PA process [No need to send other 3D parameters except **vbvTransECI**] *Note: Bank's approval is required for non-3D transactions, please contact us for further details.
vbvCHReturnCode	Number	Verify Enrollment Return Code - Set to "0" if the *enrolled* value obtained in MPI VE Response Message is "Y" - Set to "1001" if the *enrolled* value obtained in MPI VE Response Message is "N"

		- Set to “-1” if the *enrolled* value is not available in MPI VE Response Message
vbvPAReturnCode	Number	Payer Authentication Return Code - Set to “0” if the *status* value obtained in MPI VE Response Message is “Y” - Set to “1000” if the *status* value obtained in MPI VE Response Message is “A” - Set to “1003” if the *status* value obtained in MPI VE Response Message is “N” - Otherwise, set to “-1”
vbvTransTime	Text in format (YYYYMMDD HH:MM:SS)	Transaction Time, *time*, MPI PA Response Message
vbvTransAuth	Text (28)	Cardholder Authentication Verification Value, *CAVV,* value in MPI PA Response Message [Must be in Base64-Encoded format]
vbvTransECI	Text (2)	Electronic Commerce Indicator, *ECI*, value in MPI PA Response Message For VISA & JCB card payment: - For Enrolled 3D VISA card [*enrolled* = Y] - Set it to the value of *ECI* obtained from MPI PA Response Message - Set it to “07” if *ECI* value is not available in MPI PA Response Message - For Not an Enrolled 3D VISA card [*enrolled* = N] - Set it to “06” - Otherwise, set it to “07” For MasterCard payment: - For Enrolled 3D MasterCard [*enrolled* = Y] - Set it to the value of *ECI* obtained from MPI PA Response Message - Set it to “00” if *ECI* value is not available in MPI PA Response Message

		- For Not an Enrolled 3D MasterCard [*enrolled* = N] - Set it to “01” - Otherwise, set it to “00 For Other credit card payment, set it to “07”
vbvCAVVAlgo	Text	CAVV Algorithm, *cavvAlgorithm*, in MPI PA Response Message
vbvXID	Text(20)	Transaction Identifier, *xid*, in MPI PA Response Message [Must contain 20 characters]
vbvMerchantID	Text	Acquirer-defined Merchant Identifier, *merID*, in MPI PA Response Message
vbvAcquirerBin	Text	Acquirer BIN, *acqBIN*, in MPI PA Response Message
vbvTransStatus	Text(1)	Transaction Status, *status*, in MPI PA Response Message - Set it to the value of *status* obtained from MPI PA Response Message - Set it to “U” if the *status* value is not available in the MPI PA Response Message
Return Parameter		
src	Number	Return bank host status code
prc	Number	Return bank host status code
Ord	Number	Bank Reference – Order id
Holder	Text	The Holder Name of the Payment Account
successcode	Number	Transaction Status: -1 - Error 0 - Transaction succeeded 1 - Transaction Failure
Ref	Text	Merchant‘s Order Reference Number
PayRef	Number	Payment Reference Number
Amt	Number (15,5)	Transaction Amount
Cur	Number (3)	Transaction Currency i.e. 344 mean HKD
AuthId	Text	Approval Code
TxTime	Text (YYYY-MM-DD HH:MI:SS.0)	Transaction Time
errMsg	Text	Error Message

All the return parameters will be concatenated as in html request format by separate with **`&`**.

Sample return string:

```
successcode=0&Ref=Test&PayRef=4780&Amt=1.0&Cur=344&prc=0&src=0&Ord=6697090&
Holder=edward&AuthId=123456&TxTime=2003-10-07 17:48:02.0&errMsg=Transaction
completed
```

Example of Source Code

As different type of programming language have different syntax, so we just propose the method to connect to our payment page. To connect, we suggest you to use server side posting:

Sample code for server post by using java:

```
// Set up the post data
String postData =
"merchantId=1&orderRef=test&amount=1&currCode=344&pMethod=VISA&epMonth=01&epYear=20
02&cardNo=4123412341234123&cardholder=Edward&remark=test”;
// Post to payment page
strResult = ServerPost.post(postData,
        https://www.paydollar.com/b2c2/eng/directPay/payComp.jsp );
// Extract the payment status from strResult
...
// Finish
************************************************************************
public class ServerPost
{
       static public String post( String ip_postData, String ip_pageUrl)
       {
              try
              {
                     String strResult = "";
                     URL url = new URL(ip_pageUrl);

                     URLConnection con = url.openConnection(); //from secure site
                     if(con instanceof com.sun.net.ssl.HttpsURLConnection){
                        ((com.sun.net.ssl.HttpsURLConnection)con).setSSLSocketFactory
                                   (SSLSocketFactory)SSLSocketFactory.getDefault());
                     }

                     con.setDoOutput(true);
                     con.setDoInput(true);
                     // Set request headers for content type and length
                     con.setRequestProperty(
                            "Content-type",
                            "application/x-www-form-urlencoded");
                     con.setRequestProperty(
                            "Content-length",
                            String.valueOf(ip_postData.length()));
                     // Issue the POST request
                     OutputStream outStream = con.getOutputStream();
                     outStream.write(ip_postData.getBytes());
                     outStream.flush();
                     // Read the response
```

```
                        InputStream inStream = con.getInputStream();

                       while (true)
                        {
                                int c = inStream.read();
                                if (c == -1)
                                        break;
                                strResult = strResult + String.valueOf((char)c);
                        }

                        inStream.close();
                        outStream.close();

                        return strResult;
                }
                catch (Exception e)
                {
                        System.out.print(e.toString());
                        return null;
                }
        }
}
```

** All the source code in this document are the property of AsiaPay (HK) Limited. Any use, modification and adaptation to the code should be reported to and approved by AsiaPay (HK) Limited. AsiaPay (HK) Limited do not have any liability in any lose to the party using the source code.*

Kick Off

After the integration has been completed, it is ready to launch your e-commerce web to serve your customers. Please copy the following **TESTING URL** for Direct Connect Server Post method:

`https://test.paydollar.com/b2cDemo/eng/directPay/payComp.jsp`

Please copy the following **PRODUCTION URL** for Direct Connect Server Post method:

`https://www.paydollar.com/b2c2/eng/directPay/payComp.jsp`

---

3 Data Feed handling

To use data feed function, merchant has to create a data feed page and inform PayDollar about the location of your page (e.g. http://www.yourdomain.com/datafeed.jsp). Merchant has to wait until PayDollar has updated the system in order to use this function.

Definition of Parameters in the output of Data Feed

Parameters	Data Type	Descriptions
Data Feed Output		
src	Number	Return bank host status code (secondary). Please refer to Appendix A for detail.
prc	Number	Return bank host status code (primary). Please refer to Appendix A for detail.
Ord	Number (40)	Bank Reference – Order id
Holder	Text	The Holder Name of the Payment Account
successcode	Number	0- succeeded, 1- failure, Others - error
Ref	Text	Merchant's Order Reference Number
PayRef	Number	PayDollar Payment Reference Number
Amt	Number (12,2)	Transaction Amount
Cur	Text (3)	Transaction Currency i.e. "344" - HKD "840" – USD "702" – SGD "156" – CNY (RMB) "392" – JPY "901" – TWD "036" – AUD "978" – EUR "826" – GBP "124" – CAD
mpsAmt	Number (12,2)	MPS Transaction Amount **Remark: For MPS Enabled only.**
mpsCur	Text (3)	MPS Transaction Currency **Remark: For MPS Enabled only.**
mpsForeignAmt	Number (12,2)	MPS Transaction Foreign Amount **Remark: For MPS Enabled only.**

<table>
<tr><td><b>mpsForeignCur</b></td><td>Text (3)</td><td>MPS Transaction Foreign Currency
<b>Remark: For MPS Enabled only.</b></td></tr>
<tr><td><b>mpsRate</b></td><td>Number (12,4)</td><td>MPS Exchange Rate: (Foreign / Base)
e.g. USD / HKD = 7.77
<b>Remark: For MPS Enabled only.</b></td></tr>
<tr><td><b>remark</b></td><td>Text (200)</td><td>A remark field for you to store additional data that will not show on the transaction web page</td></tr>
<tr><td><b>AuthId</b></td><td>Text</td><td>Approval Code</td></tr>
<tr><td><b>eci</b></td><td>Text (2)</td><td>ECI value (for 3D enabled Merchants)
<table>
<tr><th colspan="2">VISA</th></tr>
<tr><th>ECI Value</th><th>Definition</th></tr>
<tr><td>05</td><td>Both cardholder and card issuing bank are 3D enabled. 3D card authentication is successful</td></tr>
<tr><td>06</td><td>Either cardholder or card issuing bank is not 3D enrolled. 3D card authentication is unsuccessful, in sample situations as:
1. 3D cardholder not enrolled
2. Card issuing bank is not 3D Secure ready</td></tr>
<tr><td>07</td><td>Authentication is unsuccessful or not attempted. The credit card is either a non-3D card or card issuing bank does not handle it as a 3D transaction</td></tr>
<tr><th colspan="2">MasterCard</th></tr>
<tr><th>ECI Value</th><th>Definition</th></tr>
<tr><td>00</td><td>Authentication is unsuccessful or not attempted. The credit card is either a non-3D card or card issuing bank does not handle it as a 3D transaction</td></tr>
<tr><td>01</td><td>Either cardholder or card issuing bank is not 3D enrolled. 3D card authentication is unsuccessful, in sample situations as:
1. 3D Cardholder not enrolled
2. Card issuing bank is not 3D Secure ready</td></tr>
<tr><td>02</td><td>Both cardholder and card issuing bank are 3D enabled. 3D card authentication is successful</td></tr>
<tr><th colspan="2">JCB</th></tr>
<tr><th>ECI Value</th><th>Definition</th></tr>
<tr><td>05</td><td>Both cardholder and card issuing bank are 3D enabled. 3D card authentication is successful</td></tr>
<tr><td>06</td><td>Either cardholder or card issuing bank is not 3D enrolled. 3D card authentication is unsuccessful, in sample situations as:
1. 3D cardholder not enrolled
2. Card issuing bank is not 3D Secure ready</td></tr>
<tr><td>07</td><td>Authentication is unsuccessful or not attempted. The credit card is either a non-3D card or card issuing bank does not handle it as a 3D transaction</td></tr>
</table>
Remark : Empty String will be sent when the transaction is rejected by PayDollar PayAlert.</td></tr>
</table>

payerAuth	Text (1)	Payer Authentication Status Y - Card is 3D-secure enrolled and authentication succeeds. N - Card is 3D-secure enrolled but authentication fails. P - 3D Secure check is pending A - Card is not 3D-secure enrolled yet U - 3D-secure check is not processed.
sourceIp	Text (15)	IP address of payer
ipCountry	Text (3)	Country of payer (e.g. HK) - if country is on high risk country list, an asterisk will be shown (e.g. MY*)
payMethod	Text (10)	Payment method (e.g. VISA, Master, Diners, JCB, AMEX)
cardIssuingCountry	Text (3)	Card Issuing Country Code (e.g. HK) - if country is on high risk country list, an asterisk will be shown (e.g. MY*) - if the card issuing country of credit card is undefined, “- -“ will be shown. Please refer to Appendix A “List of Country Code” for detail
secureHash	Text (40)	Secure hash is used to authenticate the integrity of the response information and the identity of PayDollar. It is calculated by hashing the combination of various response parameters and the Secure Hash Secret. *Applies to merchants who registered this function only. For more information, please refer to section 4.

The data feed page must meet the following requirement:

- Print ‘**OK**’ in HTML when data captured (ACK message)
- **Make Sure to Print ‘OK’ for acknowledge to our system first then do the rest of your system process, if something wrong with your system process (i.e. download photo, ring tone problem) you can send a void request to our system, for more details please refer to our API guide and contact our technical staff.**

Please note that the system only supports either port 80 (HTTP) or 443 (HTTPS) for the data feed page location. And make sure the data feed page location is externally accessible, so that our server can call the data feed page.

* Since the system will read from the data feed page for the word ‘OK’ to determine whether the (data feed) message is delivered or not, if this word does not return successfully, the system will assume the data feed is lost.

Data Feed Setup

For **testing environment**, you can contact our service team for setup. Please indicate your testing merchant ID and testing data feed URL. Once received the request our service team will set up the data feed for you.

For **production environment**, you may fill in the Merchant Account Maintenance Form in the Support section of Merchant admin page. You may return the completed form via fax or email.

Sample Data Feed Page

The following is a sample data feed page in JSP.

```
<%@ page language="java" %>
<%
     String successCode = request.getParameter("successcode");
     String payRef = request.getParameter("PayRef");
     String Ref = request.getParameter("Ref");

    // Print out 'OK' to notify us you have received the payment result
    out.print("OK");

     if ( successCode.equals("0") )
     {
          // Transaction Accepted
         // *** Add the Security Control here, to check the currency, amount with the
         // *** merchant's order reference from your database, if the order exist then
         // *** accepted otherwise rejected the transaction.

         //  Update your database for Transaction Accepted and send email or notify your
         //   customer.
          ....

          // In case if your database or your system got problem, you can send a void
          transaction request. See API guide for more details
     }
     else
     {
          // Transaction Rejected
          // Update your database for Transaction Rejected
          .....
     }

%>
```

The following is a sample data feed page in ASP.

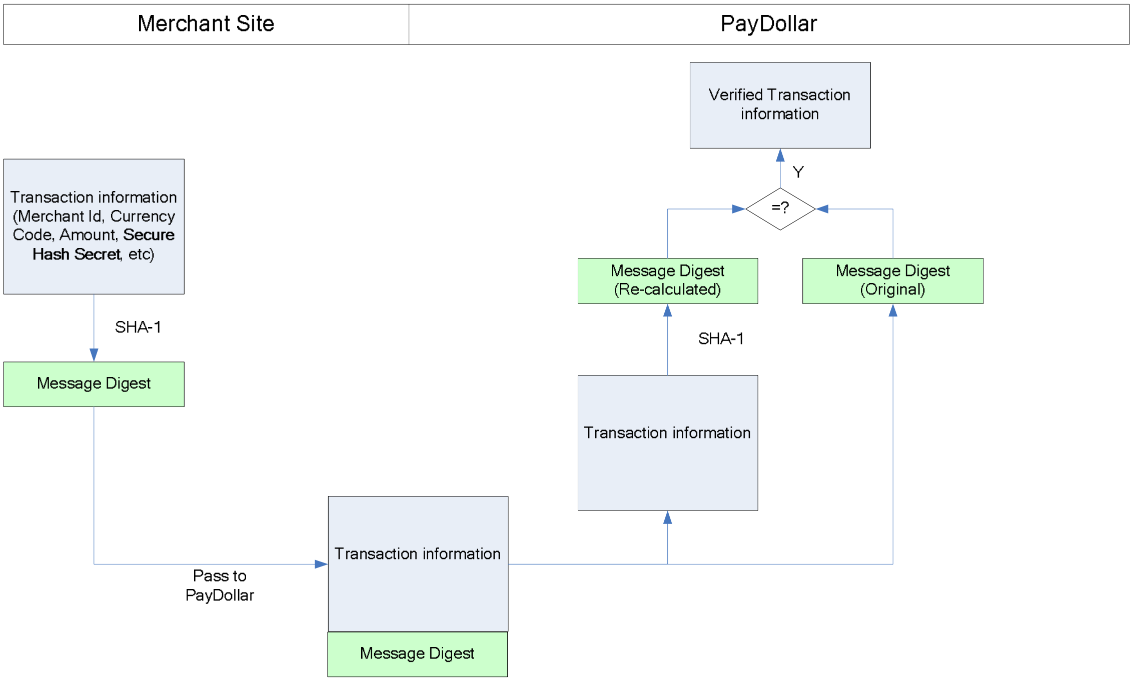
```
<%@ Language = "VBScript" %>
<%
    Dim successCode
    Dim payRef
    Dim Ref

    successCode = Request.Form("successcode")
    payRef = Request.Form("PayRef")
    Ref = Request.Form("Ref")

    ' Print out 'OK' to notify us you have received the payment result
    Response.write("OK")

    If successCode = "0" Then
        ' Transaction Accepted
        ' *** Add the Security Control here, to check the currency, amount with the
        ' *** merchant’s order reference from your database, if the order exist then
        ' *** accepted otherwise rejected the transaction.

        ' Update your database for Transaction Accepted and send email or notify your
        ' customer.
        .....

        ' In case if your database or your system got problem, you can send a void
        ' transaction request. See API guide for more details
    Else
        ' Transaction Rejected
        ' Update your database for Transaction Rejected
        .....
    End If
%>
```

4 Transaction security by Secure Hash

Introduction

The purpose of Secure Hash is to enhance the transaction message communication security between merchant site and PayDollar. By employing this technology, the integrity of the information and the identity of the signatory can be authenticated with industry standard.

Basic flow of Secure Hash


Merchant Site
PayDollar
Verified Transaction information
Y
=?
Transaction information (Merchant Id, Currency Code, Amount, Secure Hash Secret, etc)
Message Digest (Re-calculated)
Message Digest (Original)
SHA-1
SHA-1
Message Digest
Transaction information
Pass to PayDollar
Transaction information
Message Digest

1. For all transaction request send from merchant site to PayDollar, secure hash should be calculated and added to the transaction request to authenticate the integrity of the transaction information and the identity of the merchant.
2. The secure hash is calculated by hashing the following parameters using SHA-1, a cryptographic hash function of industry standard.
   - Merchant ID
   - Merchant Reference Number
   - Currency Code
   - Amount
   - Payment Type
   - Secure Hash Secret – Assigned by PayDollar to merchant.
3. When the transaction request is received, PayDollar verifies the transaction by comparing the secure hash submitted by merchant and the secure hash re-calculated by other input parameters.

If both values are the same, existing payment flow will follows. Or else, the payment request will be dropped. PayDollar will send out an email to notify the merchant through the operation contact.

4. After the transaction is completed, PayDollar will send out datafeed to merchant site. A secure hash will also be calculated by hashing the following parameters using SHA-1,
    - Src
    - Prc
    - Success Code
    - Merchant Reference Number
    - PayDollar Reference Number
    - Currency Code
    - Amount
    - Payer Authentication Status
    - Secure Hash Secret
5. After receiving the datafeed, merchant is suggested to verify the information by comparing the secure hash posted by PayDollar and the secure hash re-calculated by other response parameters. If the values doesn't matched, the datafeed may have been tampered within the redirection process and you are suggested to do further investigation before confirmation the order.

Client library provided by PayDollar

Client library is provided by PayDollar to facilitate the secure hash generation and verification process. It supports common programming language including Java, PHP, ASP, ASP.NET. The following function calls are supported.

Functions	Parameters	Descriptions
generateSecureHash	Input - Merchant ID - Merchant Reference Number - Currency Code - Amount - Payment Type - Secure Hash Secret Output - Secure Hash String	Create a secure hash using the input parameters and Secure Hash Secret. (The result secure hash should be included in the payment parameter send to PayDollar.)
verifyDatafeed	Input - Src - Prc	Verify the parameters passed from PayDollar using input parameters and Secure Hash Secret.

	- Success Code - Merchant Reference Number - PayDollar Reference Number - Currency Code - Amount - Payer Authentication Status - Secure Hash Secret - Secure Hash from PayDollar Output - True/False	(If the result of the output is true, it is verified that the result is sent from PayDollar and it is safe to trust the result.)

*Please login to PayDollar Merchant Administration Tools and download client library with sample code under Support → Developer Corner.

Generating and verifying Secure Hash manually

Merchant may also generate and verify secure hash manually, without using the client library provided by PayDollar. The following diagrams list out the exact algorithm,

Generate Secure Hash

1. Create the signing data string.
   Signing data string = Merchant ID + “|” + Merchant Reference + “|” + Currency Code + “|” + Amount + “|” + Payment Type + “|” + Secure Hash Secret
2. Secure Hash = SHA-1(Signing data string)

*SHA-1 is the original 160-bit hash function.

Example of Secure Hash Secret

gMAVIEGVpqHvxoNEqbrZRuBDFT1B0icW

Example of Signing data string

56100908|1280204670187|344|10|N|gMAVIEGVpqHvxoNEqbrZRuBDFT1B0icW

Example of Secure Hash

13068c0ef09139ea711d36bde16785a2d30b9a30

Verifying Secure Hash from PayDollar datafeed

1. Create the verify data string.

   Verify data string = Src + “|” + Prc + “|” + Success Code + “|” + Merchant Reference Number + “|” + PayDollar Reference Number + “|” + Currency Code + “|” + Amount + “|” + Payer Authentication Status + “|” + Secure Hash Secret
2. Verify Secure Hash = SHA-1 (Verify data string)
3. Extract the secure hash from PayDollar datafeed.
4. Compare the output from step 2 and step 3. If they are equals, return True, else return False.

*SHA-1 is the original 160-bit hash function.

Enable Secure Hash function of your merchant account

a) Please contact PayDollar Service Department (service@paydollar.com) to enable the Secure Hash function of your merchant account.

b) You may retrieve the Secure Hash Secret of the merchant account by accessing to the Merchant Administration Interface, “Profile” → “Payment Information”. The Secure Hash Secret must be kept safely for the function to be effective.

c) The Secure Hash Secret will be changed every 2 years to enhance the level of security.

d) Once this function is enabled, a valid Secure Hash should be included in all transaction requests. All transaction without valid Secure Hash will be dropped by PayDollar.

e) You may download client library with sample code under,

PayDollar Merchant Administration Tools → Support → Developer Corner.

5 Multi-Currency Processing Service

Introduction

PayDollar by Multi-Currency Processing Service (MPS) is an integrated e-payment transaction processing service that allows your online business of any size to securely accept real-time credit card payments from overseas cardholders and offer them the choice to pay for their goods and services in their billing currency, whilst merchants continue to be settled for transactions in their base currency.

Multi-Currency Processing Service (MPS) provides three different modes as below:

- Simple Currency Conversion (SCP)
- Multi-Currency Pricing (MCP)
- Dynamic Currency Conversion (DCC) -- **Coming soon**

Multi-Currency Processing Service (MPS) facilitates merchant to connect to our network with great flexibility. Merchant can choose the following integration method.

- Client Post through Browser (e.g. Shopping Cart)

Definition:

“**Foreign Currency**” means those non-based currency for which the Program is available to merchant from time to time supported and advised by AsiaPay.

“**Base Currency**” means the currency in which the merchant is settled for payment transactions by its acquirer.

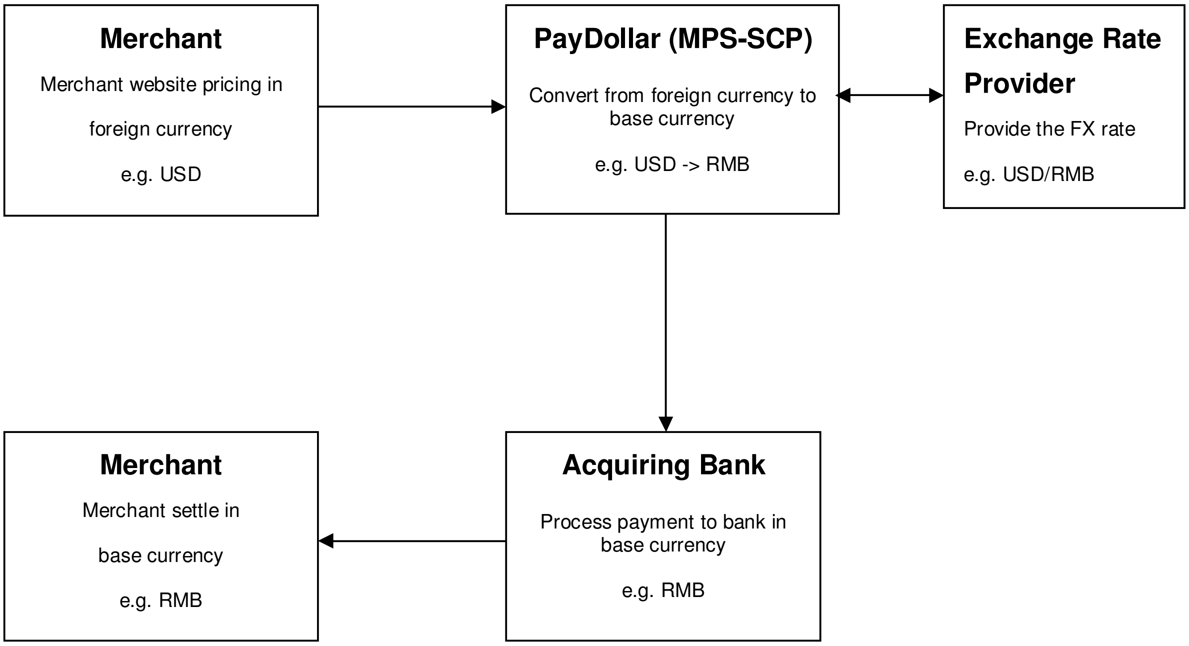
“**Conversion Rate**” means the foreign currency exchange rate derived by a recognized supplier.

Simple Currency Conversion (SCP)

Simple Currency Conversion (SCP) is a value added e-payment processing service that allows your online business to securely accept real-time credit card payments from overseas customers with **foreign currencies in pricing** while offering them to pay for goods and services in **your preferred currency.** And, your business can continue to collect settlement in base currency as usual.

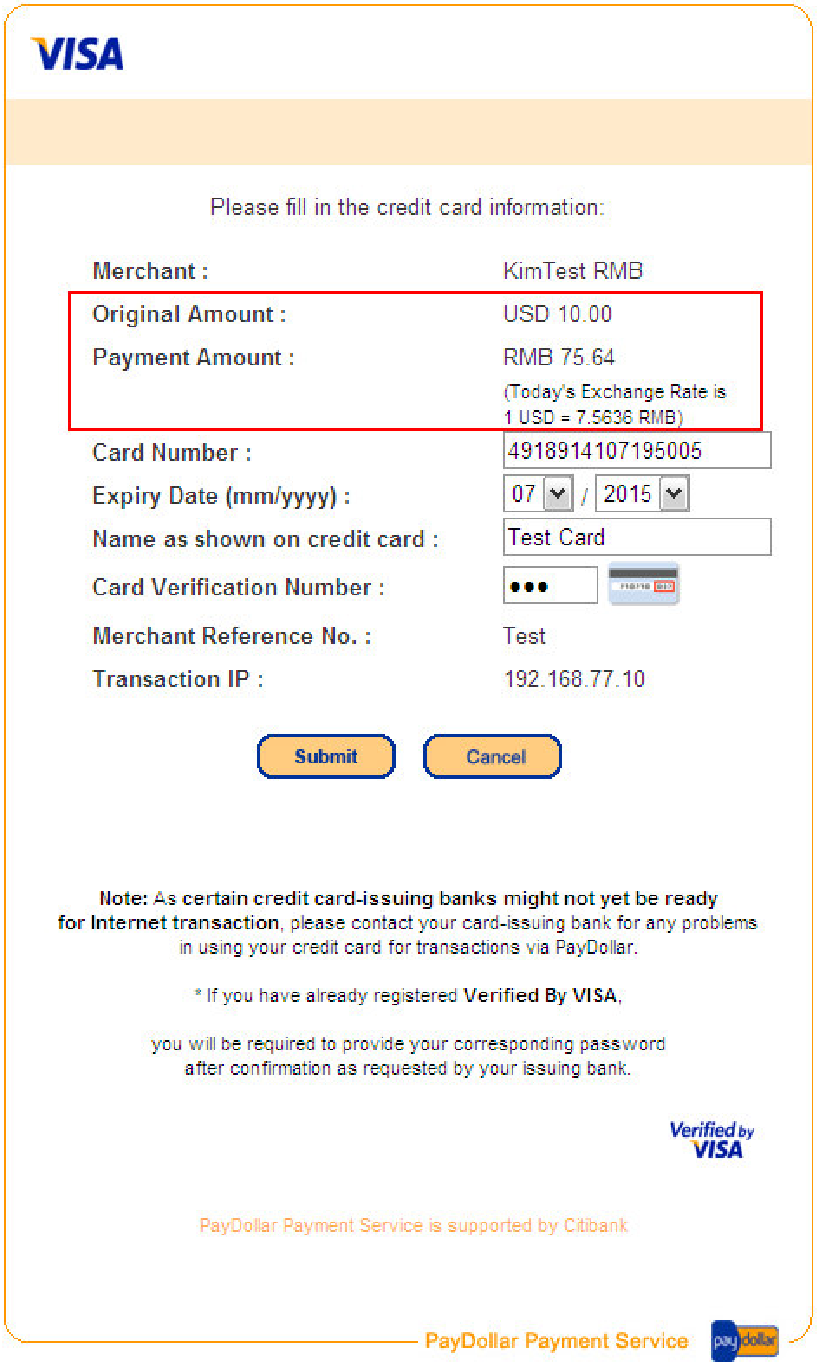
Simple Currency Conversion (SCP) will convert the foreign currency that posted by merchant to the base currency of merchant according to the conversion rate obtained from our exchange rate provider. After that, the foreign amount, base amount and the exchange rate will be shown on the Paydollar's payment input page.

Transaction Flow


Merchant
Merchant website pricing in foreign currency
e.g. USD
PayDollar (MPS-SCP)
Convert from foreign currency to base currency
e.g. USD -> RMB
Exchange Rate Provider
Provide the FX rate
e.g. USD/RMB
Merchant
Merchant settle in base currency
e.g. RMB
Acquiring Bank
Process payment to bank in base currency
e.g. RMB

Simple Currency Conversion (SCP) Sample Transaction Screen

Payment Amount Conversion and Account Input Page:



Please fill in the credit card information:

Merchant :	KimTest RMB
Original Amount :	USD 10.00
Payment Amount :	RMB 75.64 (Today's Exchange Rate is 1 USD = 7.5636 RMB)
Card Number :	4918914107195005
Expiry Date (mm/yyyy) :	07 / 2015
Name as shown on credit card :	Test Card
Card Verification Number :	●●●
Merchant Reference No. :	Test
Transaction IP :	192.168.77.10

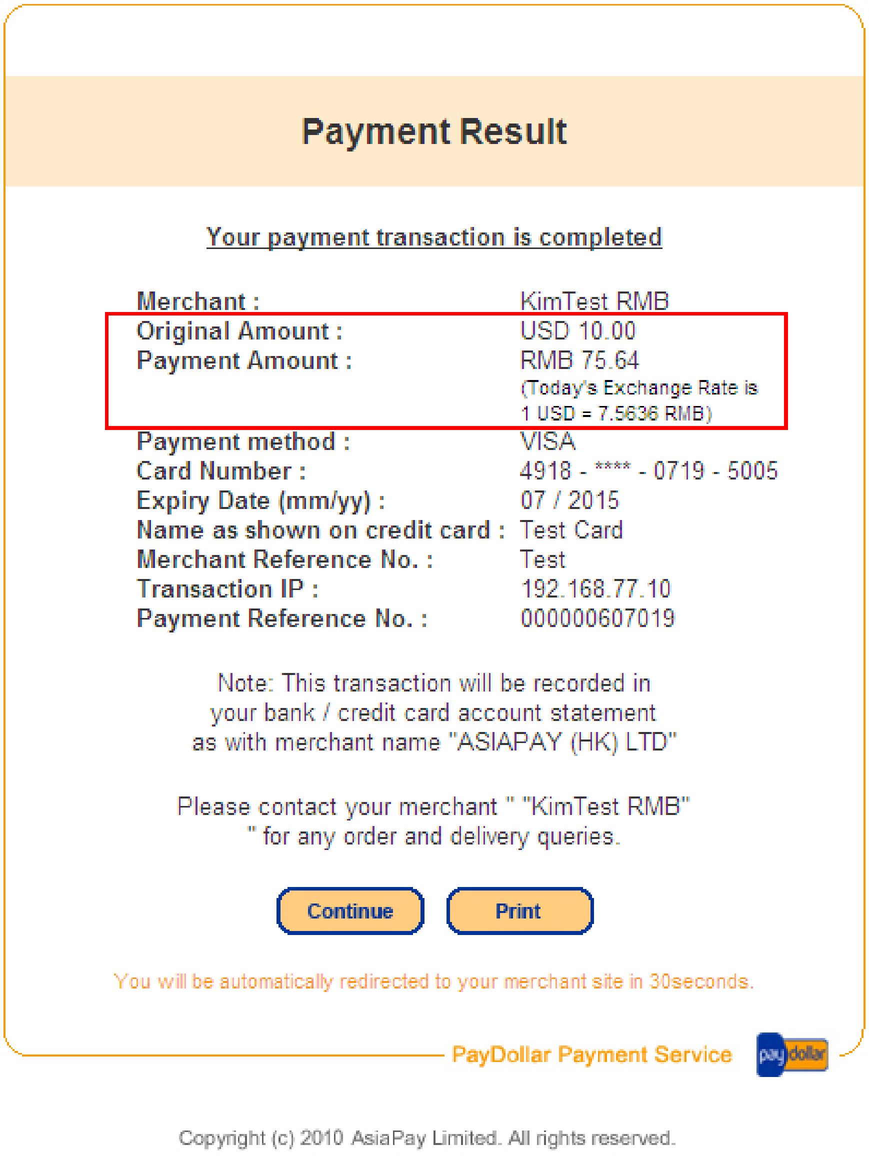
Note: As certain credit card-issuing banks might not yet be ready for Internet transaction, please contact your card-issuing bank for any problems in using your credit card for transactions via PayDollar.

* If you have already registered **Verified By VISA**,

you will be required to provide your corresponding password after confirmation as requested by your issuing bank.

Verified by VISA
PayDollar Payment Service is supported by Citibank
PayDollar Payment Service

Payment Result Page:



Payment Result

Your payment transaction is completed

Merchant :	KimTest RMB
Original Amount :	USD 10.00
Payment Amount :	RMB 75.64 (Today's Exchange Rate is 1 USD = 7.5636 RMB)
Payment method :	VISA
Card Number :	4918 - **** - 0719 - 5005
Expiry Date (mm/yy) :	07 / 2015
Name as shown on credit card :	Test Card
Merchant Reference No. :	Test
Transaction IP :	192.168.77.10
Payment Reference No. :	000000607019

Note: This transaction will be recorded in
your bank / credit card account statement
as with merchant name "ASIAPAY (HK) LTD"

Please contact your merchant " "KimTest RMB"
" for any order and delivery queries.

You will be automatically redirected to your merchant site in 30seconds.

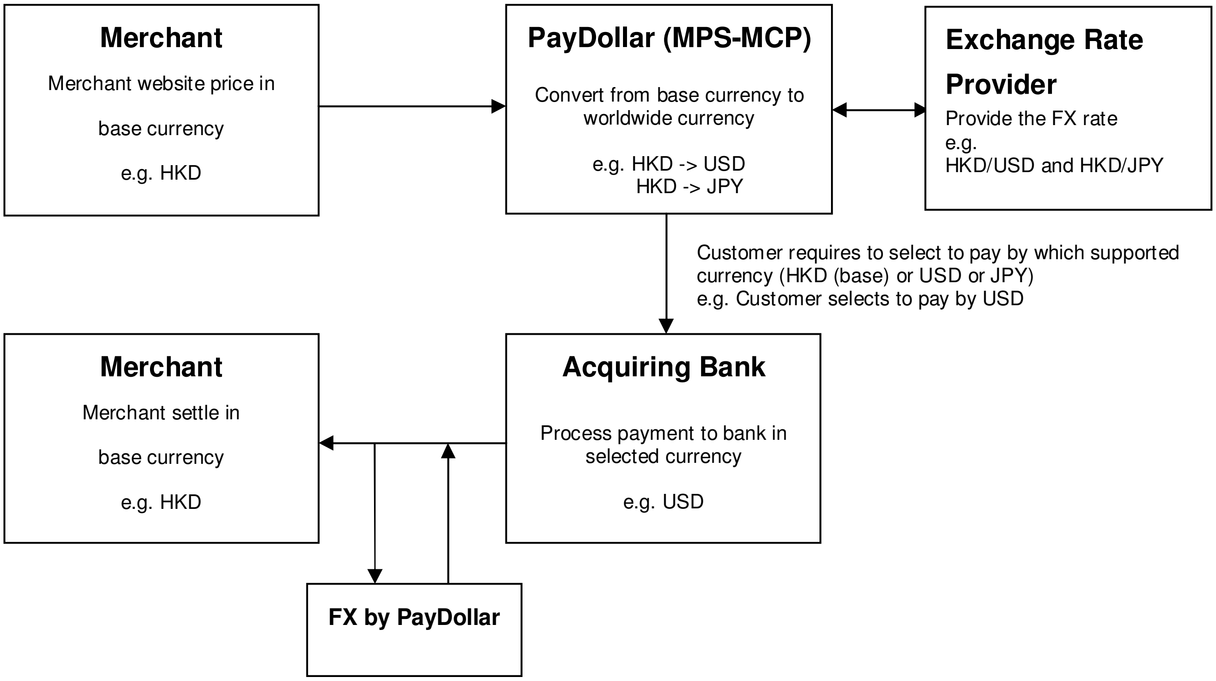
PayDollar Payment Service

Multi Currency Pricing (MCP)

Multi-Currency Pricing (MCP) is a value added e-payment processing service that allows your online business to securely accept real-time credit card payments from overseas customers while offering them the choice to pay for goods and services in **merchant base currency** or **other worldwide currencies (e.g. USD)**. And, your business can continue to collect settlement in local currency as usual.

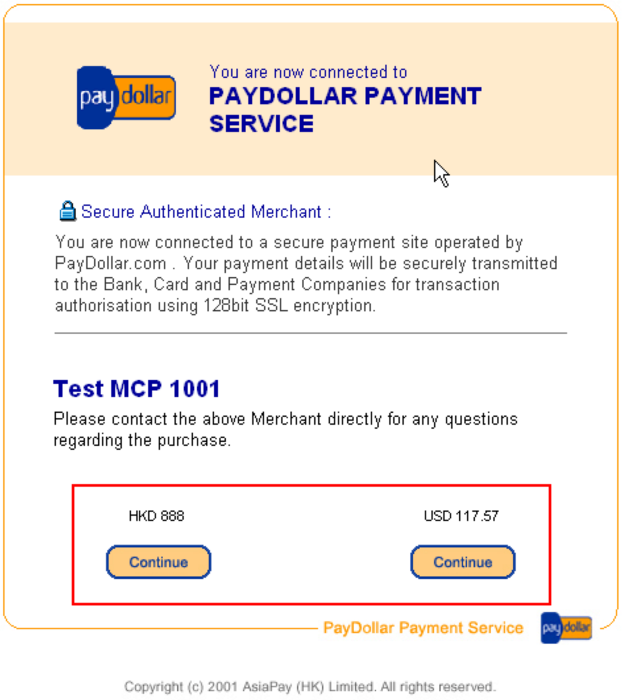
Multi-Currency Pricing (MCP) will translate the base currency that posted by merchant to the worldwide currency according to the conversion rate. After that, customer can select one of the currencies for payment.

Transaction Flow


Merchant
Merchant website price in
base currency
e.g. HKD
PayDollar (MPS-MCP)
Convert from base currency to worldwide currency
e.g. HKD -> USD
HKD -> JPY
Exchange Rate Provider
Provide the FX rate
e.g.
HKD/USD and HKD/JPY
Customer requires to select to pay by which supported currency (HKD (base) or USD or JPY)
e.g. Customer selects to pay by USD
Merchant
Merchant settle in
base currency
e.g. HKD
Acquiring Bank
Process payment to bank in selected currency
e.g. USD
FX by PayDollar

Multi-Currency Pricing (MCP) Sample Transaction Screen

MCP Payment Selection Page:



You are now connected to
PAYDOLLAR PAYMENT SERVICE

Secure Authenticated Merchant :

You are now connected to a secure payment site operated by PayDollar.com . Your payment details will be securely transmitted to the Bank, Card and Payment Companies for transaction authorisation using 128bit SSL encryption.

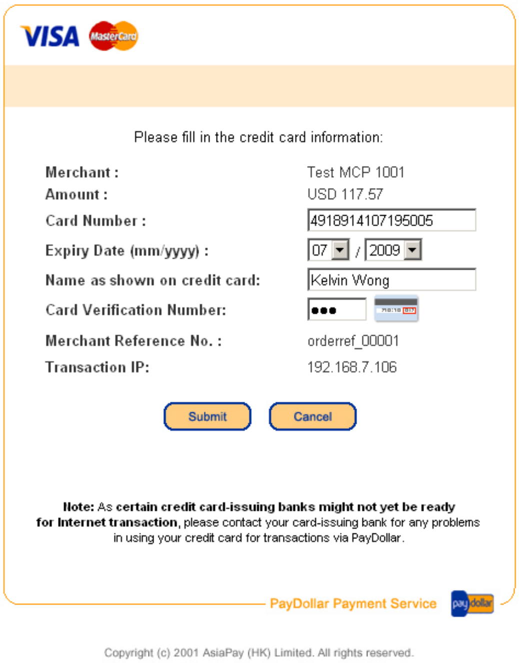
Test MCP 1001

Please contact the above Merchant directly for any questions regarding the purchase.

HKD 888	USD 117.57

Copyright (c) 2001 AsiaPay (HK) Limited. All rights reserved.

Payment Account Input Page:



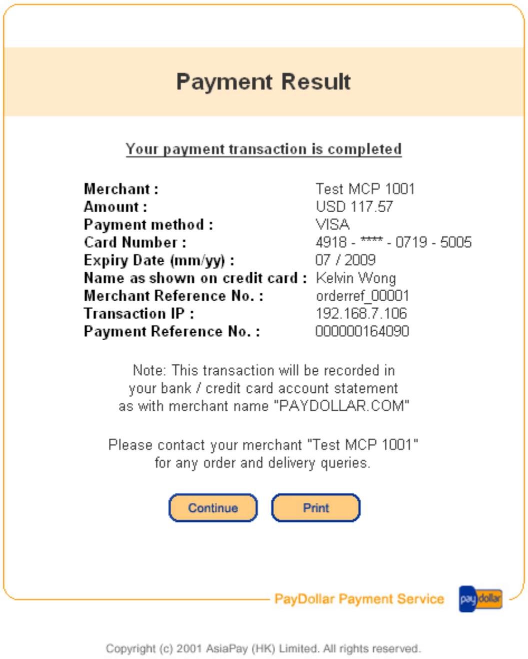
Please fill in the credit card information:

Merchant :	Test MCP 1001
Amount :	USD 117.57
Card Number :	4918914107195005
Expiry Date (mm/yyyy) :	07 / 2009
Name as shown on credit card:	Kelvin Wong
Card Verification Number:	●●●
Merchant Reference No. :	orderref_00001
Transaction IP:	192.168.7.106

Note: As **certain credit card-issuing banks might not yet be ready for Internet transaction**, please contact your card-issuing bank for any problems in using your credit card for transactions via PayDollar.

Copyright (c) 2001 AsiaPay (HK) Limited. All rights reserved.

Payment Result Page:



Payment Result

Your payment transaction is completed

Merchant :	Test MCP 1001
Amount :	USD 117.57
Payment method :	VISA
Card Number :	4918 - **** - 0719 - 5005
Expiry Date (mm/yy) :	07 / 2009
Name as shown on credit card :	Kelvin Wong
Merchant Reference No. :	orderref_00001
Transaction IP :	192.168.7.106
Payment Reference No. :	000000164090

Note: This transaction will be recorded in
your bank / credit card account statement
as with merchant name "PAYDOLLAR.COM"

Please contact your merchant "Test MCP 1001"
for any order and delivery queries.

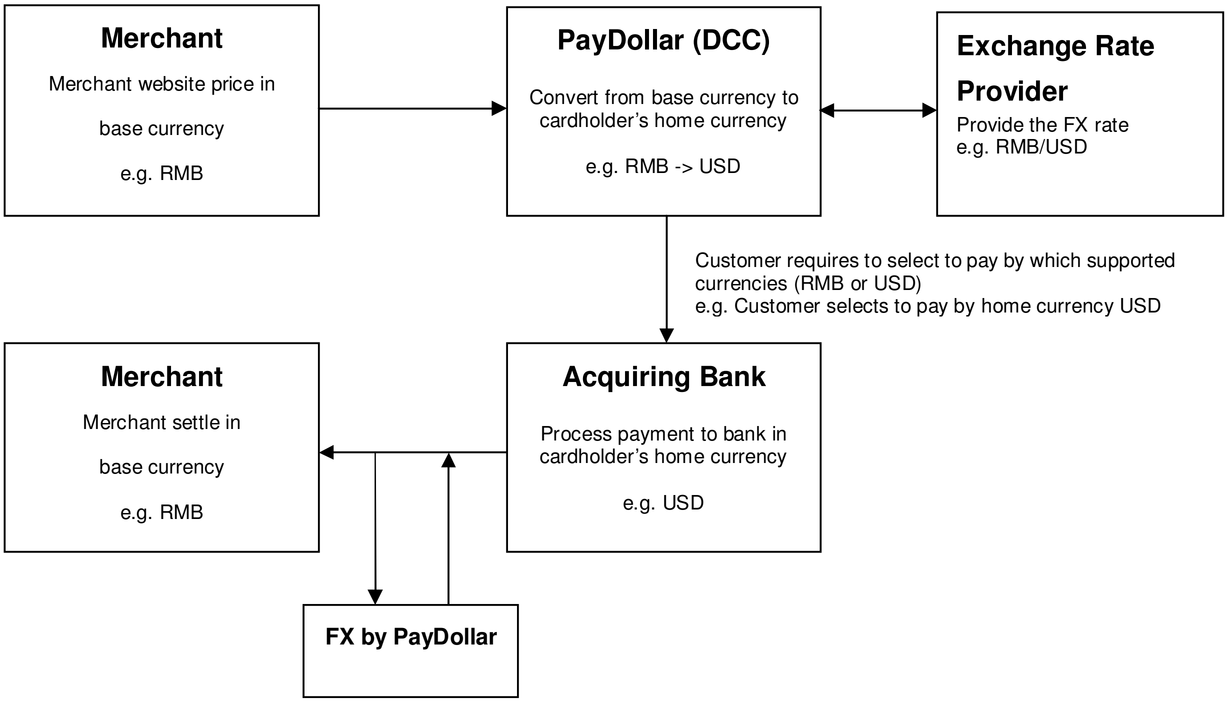
Copyright (c) 2001 AsiaPay (HK) Limited. All rights reserved.

Dynamic Currency Conversion (DCC)

Dynamic Currency Conversion (DCC) is a value added e-payment processing service that allows your online business to securely accept real-time credit card payments from overseas customers while offering them the choice to pay for goods and services in the **merchant base currency** or **cardholder's home currency**. This payment process allows the merchant to show the value of the transaction in the cardholder's home currency.

Dynamic Currency Conversion (DCC) will translate the base currency that posted by merchant to the cardholder's home currency according to the conversion rate. After that, customer can select one of the currencies for payment.

Transaction Flow


Merchant
Merchant website price in
base currency
e.g. RMB
PayDollar (DCC)
Convert from base currency to
cardholder's home currency
e.g. RMB -> USD
Exchange Rate
Provider
Provide the FX rate
e.g. RMB/USD
Customer requires to select to pay by which supported
currencies (RMB or USD)
e.g. Customer selects to pay by home currency USD
Merchant
Merchant settle in
base currency
e.g. RMB
Acquiring Bank
Process payment to bank in
cardholder's home currency
e.g. USD
FX by PayDollar

6 Functions of Merchant API

Introduction of API functions

There are totally four functions provided:-

- Capture Authorized Payment
- Void Accepted Payment
- Request Refund Accepted Payment
- Query Payment Status

To connect to our system, you need to post the required parameters by HTML form posting to our merchant api web page and then get back the processing result from that page. You can implement it by server-side html post.

- URL of Testing Platform:
  https://test.paydollar.com/b2cDemo/eng/merchant/api/orderApi.jsp
- URL of Production Platform:
  https://www.paydollar.com/b2c2/eng/merchant/api/orderApi.jsp

Beside, a set of API login ID and password will be assigned to your merchant account for accessing this API function. And it can be obtained from us by sending a request email or directly contact us.

Capture Authorized Payment

The aim of this function is to capture the authorized payment.

Definition of Parameters in the Integration Page

Input /Return	**Parameters (Required Fields are in Bold typeface)** *Case Sensitive	**Data Type**	**Expected Value**	**Descriptions**
Input	**merchantId**	Number		The merchant ID we provide
	loginId	Text (30)		The loginId of merchant API
	password	Text (15)		The password of merchant API
	actionType		"Capture"	The action type
	payRef	Text (35)		Payment Reference Number
	amount	Number (12,2)		The amount you want to capture (must be less than or equal to the original amount)
Return	resultCode	Number	{"0","-1"}	0 - Request Successfully -1 – Request Failed
	orderStatus	Text(20)		The new order status after successfully request
	ref	Text		Merchant's Order Reference Number
	payRef	Number		PayDollar transaction reference
	amt	Number (12,2)		Transaction Amt
	cur	Number (3)		Transaction Currency i.e. "344" - HKD "840" – USD "702" – SGD "156" – CNY (RMB) "392" – JPY "901" – TWD
	errMsg	Text		Error Message

All the return parameters will be concatenated as in html request format by separate with **&**

Sample return string:

```
resultCode=0&orderStatus=Accepted&ref=Test&payRef=4780&amt=1.0&cur=344&errM
sg=Capture Successfully.
```

Void Accepted Payment

The aim of this function is to void the accepted payment before settlement. It can be done only before our settlement time.

Definition of Parameters in the Integration Page

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide to you
	loginId	Text (30)		The loginId of using merchant API
	password	Text (15)		The password of using merchant API
	actionType		"Void"	The action type
	payRef	Text (35)		Payment Reference Number
Return	resultCode	Number	{"0","-1"}	0 - Request Successfully -1 – Request Failed
	orderStatus	Text(20)		The new order status after successfully request
	ref	Text		Merchant's Order Reference Number
	payRef	Number		PayDollar transaction reference
	amt	Number (12,2)		Transaction Amt
	cur	Number (3)		Transaction Currency i.e. "344" - HKD "840" – USD "702" – SGD "156" – CNY (RMB) "392" – JPY "901" – TWD
	errMsg	Text		Error Message

All the return parameters will be concatenated as in html request format by separate with **&**

Sample return string:

`resultCode=0&orderStatus=Voided&ref=Test&payRef=4780&amt=1.0&cur=344&errMsg=Void Successfully.`

Request Refund for Accepted Payment

The aim of this function is to request refund for accepted payment before 14 days.

Definition of Parameters in the Integration Page

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide to you
	loginId	Text (30)		The loginId of using merchant API
	password	Text (15)		The password of using merchant API
	actionType		"RequestRefund"	The action type
	payRef	Text (35)		Payment Reference Number
(Optional Input)	**amount**	Number (12,2)		The amount you want to refund (must be less than or equal to the original amount)
Return	resultCode	Number	{"0","-1"}	0 - Request Successfully -1 – Request Failed
	orderStatus	Text(20)		The new order status after successfully request
	ref	Text		Merchant's Order Reference Number
	payRef	Number		PayDollar transaction reference
	amt	Number (12,2)		Transaction Amt
	cur	Number (3)		Transaction Currency i.e. "344" - HKD "840" – USD "702" – SGD "156" – CNY (RMB) "392" – JPY "901" – TWD
	errMsg	Text		Error Message

All the return parameters will be concatenated as in html request format by separate with **&**

Sample return string:

`resultCode=0&orderStatus=RequestRefund&ref=Test&payRef=4780&amt=1.0&cur=344 &errMsg=Request successfully and we will process it later.`

Query payment status

The aim of this function is to query the payment status on an order by either Merchant Reference Number or Payment Reference Number with XML

Definition of Parameters in the Integration Page

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide to you
	loginId	Text (30)		The loginId of using merchant API
	password	Text (15)		The password of using merchant API
	actionType		"Query"	The action type
	orderRef	Text (35)		Merchant Reference Number
	payRef	Text (35)		Payment Reference Number
Return	orderStatus	Text(20)		The new order status after successfully request
	ref	Text		Merchant's Order Reference Number
	payRef	Number		PayDollar transaction reference
	mpsMode	Text(3)		The Multi – Currency Processing Service (MPS) Mode: "NIL" or not provide – Disable MPS (No currency conversion) "SCP" – Enable MPS with 'Simple Currency Conversion' "DCC" – Enable MPS with 'Dynamic Currency Conversion' "MCP" – Enable MPS with 'Multi

				Currency Pricing’
	amt	Number (12,2)		Transaction Amt
	cur	Number (3)		Transaction Currency i.e. “344” - HKD “840” – USD “702” – SGD “156” – CNY (RMB) “392” – JPY “901” – TWD
	src	Number		Bank Return Status code 1
	prc	Number		Bank Return Status code 2
	ord	Number		Bank Reference Number
	holder	Text		The Holder Name of the Payment Account
	sourceIp	Text (15)		IP address of payer
	ipCountry	Text (3)		Country of payer (e.g. HK) - if country is on high risk country list, an asterisk will be shown (e.g. MY*)
	payMethod	Text (10)		Payment method (e.g. VISA, Master, JCB, AMEX)
	cardIssuingCountry	Text (3)		Card Issuing Country Code (e.g. HK) - if country is on high risk country list, an asterisk will be shown (e.g. MY*) - if the card issuing country of credit card is undefined, “- -“ will be shown. Please refer to Appendix A “List of Country Code” for detail
	mpsAmt	Number (12,2)		MPS Transaction Amount **Remark: For MPS Enable only.**
	mpsCur	Text (3)		MPS Transaction Currency **Remark: For MPS Enable**

				only.
	mpsForeignAmt	Number (12,2)		MPS Transaction Foreign Amount **Remark: For MPS Enable only.**
	mpsForeignCur	Text (3)		MPS Transaction Foreign Currency **Remark: For MPS Enable only.**
	mpsRate	Number (12,4)		MPS Exchange Rate: (Foreign / Base) e.g. USD / HKD = 7.77 **Remark: For MPS Enable only.**
	errMsg	Text		Error Message

All the return parameters will be in XML format

Sample return XML file:

```
<records>
    <record>
        <orderStatus>Accepted</orderStatus>
        <ref>Test</ref>
        <payRef>1390545</payRef>
        <mpsMode>NIL</mpsMode>
        <amt>1</amt>
        <cur>344</cur>
        <prc>0</prc>
        <src>0</src>
        <ord>00004295104</ord>
        <holder>Holder Name</holder>
        <sourceIp>192.168.77.10</sourceIp>
        <ipCountry>HK</ipCountry>
        <payMethod>VISA</payMethod>
        <cardIssuingCountry>HK</cardIssuingCountry>
        <mpsAmt></mpsAmt>
        <mpsCur></mpsCur>
        <mpsForeignAmt></mpsForeignAmt>
        <mpsForeignCur></mpsForeignCur>
        <mpsRate></mpsRate>
        <errMsg>Query Successfully</errMsg>
    </record>

    <!-- more records … … … -->
</records>
```

Settlement report request

The aim of this function is to generate settlement report.

Definition of Parameters in the Integration Page

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide to you
	loginId	Text (30)		The loginId of using merchant API
	password	Text (15)		The password of using merchant API
	startDate	Number(14)	DDMMYYYY hhmmss	Report Start Date
	endDate	Number(14)	DDMMYYYY hhmmss	Report End Date
(Optional Input)	**queryType**	Text(1)	O/S	Report Type: O - Use date/time of the authorization or sales transaction to generate report (default) S - Use date/time of the settlement transaction to generate report and only query transactions that settleflag is T
Return	authdate	Number(14)		The date/time of the authorization or sales transaction
	capturedate	Number(14)		The date/time of the captured (after authorize) transaction
	batchid	Number(10)		Settlement batch ID
	settledate	Number(14)		The date/time of the settlement transaction
	payref	Number (14)		Unique number in Payment platform

	merref	Text(30)		Merchant order reference number
	authid	Text(6)		Approval code
	cur	Number		Bank Return Status code 2
	amt	Number(12,2)		Transaction Amount
	orderstatus	Text		The Holder Name of the Payment Account
	terminal	Text(100)		Bank Terminal id
	bankmid	Text(100)		Bank Merchant id
	settleflag	boolean	{“T”,”F”}	Settled or not
	src	Text		Return Payment Manager Status
	prc	Text		Return Payment Manager Status
	errMsg	Text		Error Message

URL of Testing Platform:

https://test.paydollar.com/b2cDemo/GenTxnXML

URL of Production Platform:

https://www.paydollar.com/b2c2/GenTxnXML

All the return parameters will be in XML format

Sample return XML file:

```
<?xml version=”1.0” encoding=’UTF-8”?>
<reports>
<report>
    <authdate>24092005223000</authdate>
    <capturedate>24092005230000</capturedate>
    <batchid>123456</batchid>
    <!-- more parameter … … … -->
</report>

<!-- more reports … … … -->
</reports>
```

Sample source code of HTML server-side posting on Java

As different type of programming language have different syntax, so we just propose one method to connect to our merchant api page. To connect, we suggest you to use server side posting:

Sample code for serverpost by using java:

```
//      SET UP THE POST DATA

String postData =
"merchantId=1&loginId=testing&password=pwd&payRef=123456&actionType=Capture&amount=
1&”;

//      POST TO PAYMENT PAGE

strResult = ServerPost.post(postData,
http://test.paydollar.com:8080/b2cDemo/eng/merchant/api/orderApi.jsp );

//      EXTRACT THE PAYMENT STATUS FROM STRRESULT

………

//      FINISH


*************************************************************************
public class ServerPost
{
        static public String post( String ip_postData, String ip_pageUrl)
        {
                try
                {
                        String strResult = "";
                        URL url = new URL(ip_pageUrl);

                        URLConnection con = url.openConnection(); //from secure site
                        if(con instanceof com.sun.net.ssl.HttpsURLConnection){
                           ((com.sun.net.ssl.HttpsURLConnection)con).setSSLSocketFactory
                                        ((SSLSocketFactory)SSLSocketFactory.getDefault());
                        }

                        con.setDoOutput(true);
                        con.setDoInput(true);

                        // Set request headers for content type and length
                        con.setRequestProperty(
                                "Content-type",
                                "application/x-www-form-urlencoded");
```

```
					con.setRequestProperty(
						"Content-length",
						String.valueOf(ip_postData.length()));

					// Issue the POST request
					OutputStream outStream = con.getOutputStream();
					outStream.write(ip_postData.getBytes());
					outStream.flush();

					// Read the response
					InputStream inStream = con.getInputStream();

					while (true)
					{
						int c = inStream.read();
						if (c == -1)
							break;
						strResult = strResult + String.valueOf((char)c);
					}

					inStream.close();
					outStream.close();

					return strResult;
		}
		catch (Exception e)
		{
					System.out.print(e.toString());
					return null;
		}
	}
}
```

7 Exceptional Transaction Handling

This section explains various scenarios of transactions, other than good and successful transactions that may occur.

A) Unsuccessful data feed

This may occur if

1. Data feed URL is wrongly set up; or
2. Connection between PayDollar and merchant server is lost; or
3. Server of either side cannot process data feed correctly.

Since the bank has already determined the transaction status, the transaction is completed. Merchant can confirm the status by

1. Log on to Merchant Administration and retrieve the corresponding transactions in Transaction Details Report; or
2. Query the transaction status by using Merchant API.

B) Unsuccessful redirection to successUrl / failUrl / cancelUrl

This may occur if

1. Wrong / invalid returned URLs are set in the integration; or
2. Connection between the customer and merchant server is lost; or
3. Customer’s computer hangs / restarts / loses power.

Since the bank has already determined the transaction status, the transaction is completed. Merchants should educate the customer to contact the merchant and confirm the transaction status with them when such case happens.

C) Incomplete 3D authentication transactions by customer

This may occur if

1. The customer closes the browser when he / she is required to enter 3D authentication information at issuer bank webpage; or
2. The customer cannot access 3D authentication page of issuer bank due to various reasons, e.g. disabled cookies.

The transaction status remains “Pending_3D”, and payer authentication status remains “P”. In PayDollar production, a schedule job is set up to change the status from “Pending_3D” to “Rejected” from time to time. The PRC / SRC pair is also updated to 3 / 9999. Data feed is also sent out in the schedule job for these unsuccessful transactions.

Sometimes customers may return to PayDollar payment page / merchant site by pressing the ‘Back’ button of the browser and try again. The same merchant reference number is used for these retry transactions. Thus merchants may receive multiple data feeds regarding transactions with the same merchant reference number, with one success transaction followed by failed transactions. Merchants can choose to ignore the fail transactions with the same merchant reference once a successful transaction has been processed.

D) Incomplete 99BILL / ALIPAY / CHINAPAY / PPS / TENPAY transactions by customer

This may occur if

1. The customer closes the browser when he / she is required to enter 99BILL / ALIPAY / CHINAPAY / PPS / TENPAY account information at respective site; or
2. The customer cannot access 99BILL / ALIPAY / CHINAPAY / PPS / TENPAY page due to various reasons, e.g. disabled cookies or 99BILL / ALIPAY / CHINAPAY / PPS / TENPAY host is down.

The transaction status remains “Pending”. In production environment a schedule job is set up in our servers to change the status from “Pending” to “Rejected” from time to time. Data feed is also sent out in the schedule job for these unsuccessful transactions.

Sometimes customers may return to PayDollar payment page by pressing the ‘Back’ button of the browser and try again. The same merchant reference number is used for these retry transactions.

Thus merchants may receive multiple data feeds regarding transactions with the same merchant reference number, with one success transaction followed by failed transactions. Merchants can choose to ignore the fail transactions with the same merchant reference once a successful transaction has been processed.

8 Frequently Asked Questions

System Setup

1. **What programming languages are supported in the Integration?**

   HTML, ASP, PHP, JSP / Servlet, and any other server side scripting languages that support HTTP protocol.

2. **Is there any consideration on firewall issues on Merchants side?**

   Merchants have to open HTTP port for data feed handling, i.e. port 80 (HTTP) / port 443 (HTTPS).

3. **Does PayDollar PayGate support any shopping cart software?**

   Technically yes. You are however required to know how to deploy the shopping cart software to work for your requirements. Samples include OSCommerce.

Common Problems

4. **During the integration I encounter the error message “Your payment service is not active.”**

   Make sure you are using the corresponding pair of merchant ID and integration URL.

   If you are using the TESTING URL (test.paydollar.com), the TESTING merchant ID, a 6-digits number, should be used. If you are using the PRODUCTION URL (www.paydollar.com), PRODUCTION merchant ID should be used, which is 4-digits number or 8-digits number.

5. **During the integration I encounter the error message “CurrCode is incorrect.”**

   One PayDollar merchant ID only allows one currency. Make sure you are using the corresponding currency for the merchant ID in the HTML form.

   To apply multi-currencies, please contact our salespersons to open additional merchant accounts.

6. **Can I make use of the calling of successUrl / failUrl / cancelUrl solely to determine the transaction status?**

   Customer may call the successUrl / failUrl / cancelUrl (with the merchant reference appended) in the browser and pretend the transaction is completed. Thus we recommend merchants to use data feed to determine the transaction status. Since PayDollar and the merchant are the only parties with the knowledge of the data feed URL, it is safe to determine the status by using the data feed.

7. **What is the difference between the parameters: Ref (orderRef), PayRef and Ord?**

   Ref (orderRef) is merchant’s own order reference number. This comes from merchant’s database or invoices.

   PayRef is PayDollar order reference number. It is unique among all orders from different merchants in PayDollar system.

   Ord is bank reference number. It is generated by acquiring banks.

To seek help on transactions, please provide PayRef (preferred) or Ref to us.

Data Feed

8. **How do I make use of the data feed?**

   You may make use of the data passed from PayDollar in the data feed to update your database records regarding the transaction. Since merchant reference can be retrieved from data feed, you can make use of this key to update the corresponding transaction records of your system(s).

9. **How to set up data feed in my merchant account?**

   For **testing environment**, you can contact our service team for setup. Please indicate your testing merchant ID and testing data feed URL. Once received the request our service team will set up the data feed for you.

   For **production environment**, you may fill in the Merchant Account Maintenance Form in the Support section of Merchant admin page. You may return the completed form via fax or email.

10. **How do I know if the data feed is set up properly?**

    You may go to transaction details in the merchant administration site, select output columns "Data Feed Ind." and "Data Feed Return" and view the transaction records. If the data feed indicator is "T", the data feed is set up properly. However, if the indicator is "F", you may look at the "Data Feed Return" column and see what the error is. You may ask our technical team to assist when it relates to data feed setup.

    In addition, you can enable "Data Feed Failure Alert by Email" in the merchant administration site > Profile > Payment Options. When it is enabled, and if there is any failure on calling Merchant's data feed, an email will be sent to Merchant's technical contact email account.

11. **What programming languages can be used in writing data feed page?**

    Any programming languages that can handle HTTP Post request parameters can be used.

12. **How do I write the data feed page? Should the data feed page display anything on the screen?**

    The data feed page contains 3 parts, receive HTTP parameters, print 'OK' and your own backend processing (e.g. update database, send email to customer, etc.).

    The data feed page is called in the back end. It should not display anything, i.e. no HTML code. However it should respond with the word 'OK' to let our server know that you have successfully received the data feed.

13. **How do I know for sure the data feed is calling from PayDollar?**

    You may check if the data feed HTTP request is calling from these IP ranges.

    Testing: 202.64.244.236 – 202.64.244.237

    Production: 203.98.136.1 – 203.98.136.30

14. **There is data feed error return**

"javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException: No trusted certificate found"

Our server cannot recognize your SSL certificate provider. Please contact our I.T. team to resolve the issues. When necessary you will be asked to provide your CA root certificate.

15. **Auto retry failed datafeed**

    (Applicable for merchants who have registered the datafeed link and retry function)

    Sometimes, you may not be able to receive the datafeed response due to reasons like internet connection issue, incorrect datafeed URL being used, etc.
    After enabling the "Auto retry failed data feed",
    Datafeed will be resent:
    (1) immediately after the original attempt is failed, and
    (2) 15 minutes after if (1) is also failed

3D-Secure Authentication

16. **Can I use FRAMES in designing our website?**
    Some card issuing banks require Cookies when entering their 3D authentication page. Full page (i.e. no FRAMES) is required in this situation.
    Furthermore, the SSL indicator should be displayed somewhere on the browser window to let customers know that the site is secure enough to enter sensitive information such as credit card number. When FRAMES is used, the SSL indicator may not appear if the page required SSL is inside the frame window.
    Therefore, it is suggested not to use FRAMES after the HTML form is submitted to PayDollar. You may consider opening another browser window to handle payment transactions.

17. **Can I ask for disabling 3D-Secure authentication for my transactions?**
    This is subject to bank's sole decision. You may be asked for providing business information in order to help the bank risk management team to evaluate your company profile before special approval. Nonetheless, you should be well aware the serious consequences of fraudulent non-3D transactions.

Data Security

18. **Can I store the credit card information of my customers?**
    We recommend our merchants NOT to store credit card information, especially credit card number, expiry date and CVV2 / CVC2. Even if absolute necessary, please ensure to encrypt the data compliant to Visa A.I.S. (Account Information Security) program. You may refer to the following webpage: http://www.visa-asia.com/ap/sea/merchants/riskmgmt/ais_what.shtml.
    Credit card information stored in PayDollar complies with Visa A.I.S. and MasterCard SDP programs.

Support

19. **Who should we contact in case we encounter problems during testing and in production?**

You can contact our I.T. team by either of the following ways.

a. Email: it@paydollar.com

b. Technical hotline: (852) – 82267981

(852) – 31731990

APPENDIX A

PayDollar Payment Response Code

PayDollar Payment Response Code is composed of the following items:

1. Primary Response Code (PRC)

The primary response code is the main response code used for identifying the authorization status of a payment transaction.

The following table provides a summary of all the response codes which may be returned:

PRC	Description
0	Success
1	Rejected by Payment Bank
3	Rejected due to Payer Authentication Failure (3D)
-1	Rejected due to Input Parameters Incorrect
-2	Rejected due to Server Access Error
-8	Rejected due to PayDollar Internal/Fraud Prevention Checking
-9	Rejected by Host Access Error

2. Secondary Response Code (SRC)

The secondary response code provides the detail description corresponding to the primary response code.

List of Response Code

Bank's Response Code

PRC	SRC	Description
1	01	Bank Decline
1	02	Bank Decline
1	03	Other
1	04	Other
1	05	Bank Decline
1	12	Other
1	13	Other
1	14	Input Error
1	19	Other
1	25	Other
1	30	Other
1	31	Other
1	41	Lost / Stolen Card
1	43	Lost / Stolen Card
1	51	Bank Decline
1	54	Input Error
1	55	Other
1	58	Other
1	76	Other
1	77	Other
1	78	Other
1	80	Other
1	89	Other
1	91	Other
1	94	Other
1	95	Other
1	96	Other
1	99	Other
1	2000	Other

Response Code From PayDollar

PRC	SRC	Description
-8	999	Other
-8	1000	Skipped transaction
-8	2000	Blacklist error
-8	2001	Blacklist card by system
-8	2002	Blacklist card by merchant
-8	2003	Black IP by system
-8	2004	Black IP by merchant
-8	2005	Invalid cardholder name
-8	2006	Same card used more than 6 times a day
-8	2007	Duplicate merchant reference no.
-8	2008	Empty merchant reference no.
-8	2011	Other
-8	2012	Card verification failed
-8	2013	Card already registered
-8	2014	High risk country
-8	2016	Same payer IP attempted more than pre-defined no. a day. Same payer IP attempted more than pre-defined no. a day.
-8	2017	Invalid card number
-8	2018	Multi-card attempt

Other Response Code

PRC	SRC	Description
0	0	Success
3	Any Number	Payer Authentication Fail
-1	-1	Input Parameter Error
-2	-2	Server Access Error
-9	-9	Host Access Error

List of Country Code

This list shows the country names and risk level of individual country code.

Country Code	Country Name	High risk
A2	Satellite Provider	
AD	Andorra	
AE	United Arab Emirates	
AF	Afghanistan	
AG	Antigua and Barbuda	
AI	Anguilla	
AL	Albania	
AM	Armenia	
AN	Netherlands Antilles	
AO	Angola	
AP	Asia/Pacific Region	
AQ	Antarctica	
AR	Argentina	
AS	American Samoa	
AT	Austria	
AU	Australia	
AW	Aruba	
AZ	Azerbaijan	
BA	Bosnia and Herzegovina	
BB	Barbados	
BD	Bangladesh	
BE	Belgium	
BF	Burkina Faso	
BG	Bulgaria	**
BH	Bahrain	
BI	Burundi	
BJ	Benin	
BM	Bermuda	
BN	Brunei Darussalam	
BO	Bolivia	
BR	Brazil	
BS	Bahamas	
BT	Bhutan	

BV	Bouvet Island	
BW	Botswana	
BY	Belarus	
BZ	Belize	
CA	Canada	
CD	Congo	
CF	Central African Republic	
CG	Congo	
CH	Switzerland	
CI	Cote D'Ivoire	
CK	Cook Islands	
CL	Chile	
CM	Cameroon	**
CN	China	
CO	Colombia	
CR	Costa Rica	
CU	Cuba	
CV	Cape Verde	
CY	Cyprus	
CZ	Czech Republic	
DE	Germany	
DJ	Djibouti	
DK	Denmark	
DM	Dominica	
DO	Dominican Republic	
DZ	Algeria	
EC	Ecuador	
EE	Estonia	
EG	Egypt	**
ER	Eritrea	
ES	Spain	
ET	Ethiopia	
EU	Europe	
FI	Finland	
FJ	Fiji	
FK	Falkland Islands (Malvinas)	
FM	Micronesia	

FO	Faroe Islands	
FR	France	
GA	Gabon	
GB	United Kingdom	
GD	Grenada	
GE	Georgia	
GF	French Guiana	
GH	Ghana	**
GI	Gibraltar	
GL	Greenland	
GM	Gambia	**
GN	Guinea	
GP	Guadeloupe	
GQ	Equatorial Guinea	
GR	Greece	
GT	Guatemala	
GU	Guam	
GW	Guinea-Bissau	
GY	Guyana	
HK	Hong Kong	
HM	Heard Island and McDonald Islands	
HN	Honduras	
HR	Croatia	
HT	Haiti	
HU	Hungary	
ID	Indonesia	**
IE	Ireland	
IL	Israel	**
IN	India	
IO	British Indian Ocean Territory	
IQ	Iraq	
IR	Iran	**
IS	Iceland	
IT	Italy	
JM	Jamaica	
JO	Jordan	
JP	Japan	

KE	Kenya	
KG	Kyrgyzstan	
KH	Cambodia	
KI	Kiribati	
KM	Comoros	
KN	Saint Kitts and Nevis	
KP	Korea	
KR	Korea	
KW	Kuwait	
KY	Cayman Islands	
KZ	Kazakstan	
LA	Lao People's Democratic Republic	
LB	Lebanon	
LC	Saint Lucia	
LI	Liechtenstein	
LK	Sri Lanka	
LR	Liberia	
LS	Lesotho	
LT	Lithuania	**
LU	Luxembourg	
LV	Latvia	
LY	Libyan Arab Jamahiriya	
MA	Morocco	**
MC	Monaco	
MD	Moldova	
MG	Madagascar	
MH	Marshall Islands	
MK	Macedonia	
ML	Mali	
MM	Myanmar	
MN	Mongolia	
MO	Macau	
MP	Northern Mariana Islands	
MQ	Martinique	
MR	Mauritania	
MS	Montserrat	
MT	Malta	

MU	Mauritius	
MV	Maldives	
MW	Malawi	
MX	Mexico	
MY	Malaysia	**
MZ	Mozambique	
NA	Namibia	
NC	New Caledonia	
NE	Niger	
NF	Norfolk Island	
NG	Nigeria	**
NI	Nicaragua	
NL	Netherlands	
NO	Norway	
NP	Nepal	
NR	Nauru	
NZ	New Zealand	
OM	Oman	
PA	Panama	
PE	Peru	
PF	French Polynesia	
PG	Papua New Guinea	
PH	Philippines	
PK	Pakistan	**
PL	Poland	
PR	Puerto Rico	
PS	Palestinian Territory	
PT	Portugal	
PW	Palau	
PY	Paraguay	
QA	Qatar	
RE	Reunion	
RO	Romania	**
RU	Russian Federation	**
RW	Rwanda	
SA	Saudi Arabia	
SB	Solomon Islands	

SC	Seychelles	
SD	Sudan	
SE	Sweden	
SG	Singapore	
SI	Slovenia	
SK	Slovakia	
SL	Sierra Leone	
SM	San Marino	
SN	Senegal	
SO	Somalia	
SR	Suriname	
ST	Sao Tome and Principe	
SV	El Salvador	
SY	Syrian Arab Republic	
SZ	Swaziland	
TC	Turks and Caicos Islands	
TD	Chad	
TF	French Southern Territories	
TG	Togo	
TH	Thailand	
TJ	Tajikistan	
TK	Tokelau	
TM	Turkmenistan	
TN	Tunisia	
TO	Tonga	
TR	Turkey	**
TT	Trinidad and Tobago	
TV	Tuvalu	
TW	Taiwan	
TZ	Tanzania	
UA	Ukraine	**
UG	Uganda	
UM	United States Minor Outlying Islands	
US	United States	
UY	Uruguay	
UZ	Uzbekistan	
VA	Holy See (Vatican City State)	

VC	Saint Vincent and the Grenadines	
VE	Venezuela	
VG	Virgin Islands	
VI	Virgin Islands	
VN	Vietnam	**
VU	Vanuatu	
WF	Wallis and Futuna	
WS	Samoa	
YE	Yemen	
YT	Mayotte	
YU	Yugoslavia	**
ZA	South Africa	
ZM	Zambia	
ZW	Zimbabwe	

- The End -