PayDollar PayGate

Integration Guide for Membership and MemberPay

v2.2

(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

Revision History

Revision	Date	Revision Description
1.0	Oct 27, 2011	First issue
1.1	Nov 16,2011	Revised Section “Client Post Through Browser”,”Direct Client Side Connection” and “Server Side Direct Connection”. “memberPay_memberId” is added into section “Generate MemberPay payment session”
1.2	Nov 22,2011	Revised Section “Client Post Through Browser” Input parameters “memberPay_memberId” is added into section” Direct Client Side Connection” and “Server Side Direct Connection”.
1.3	Dec 5,2011	Input parameter “securityCode” is added into “Server Side Direct Connection” Optional parameter “status” is changed to “acctStatus” in section “Add Membership”
1.4	Mar 23,2012	New Membership API “Verify Membership”
1.5	Jun 07,2012	Static Token flow is added
1.6	Jul 04,2012	Update table of contents
1.7	Oct 31,2012	Update sample of “statictoken” Remove Membership API “Verify Membership”
1.8	Mar 21,2014	Update MemberPay API “QueryAccount”
1.9	May 29,2014	“memberpay_token” for Client Post Through Browser
2.0	Jul 03,2014	Static token can be query
2.1	Nov 19,2014	Added “StaticToken Mode” flow diagram
2.2	Dec 08,2015	Correct the API input parameters

Table of Contents

1 Connection Method

1.1 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.

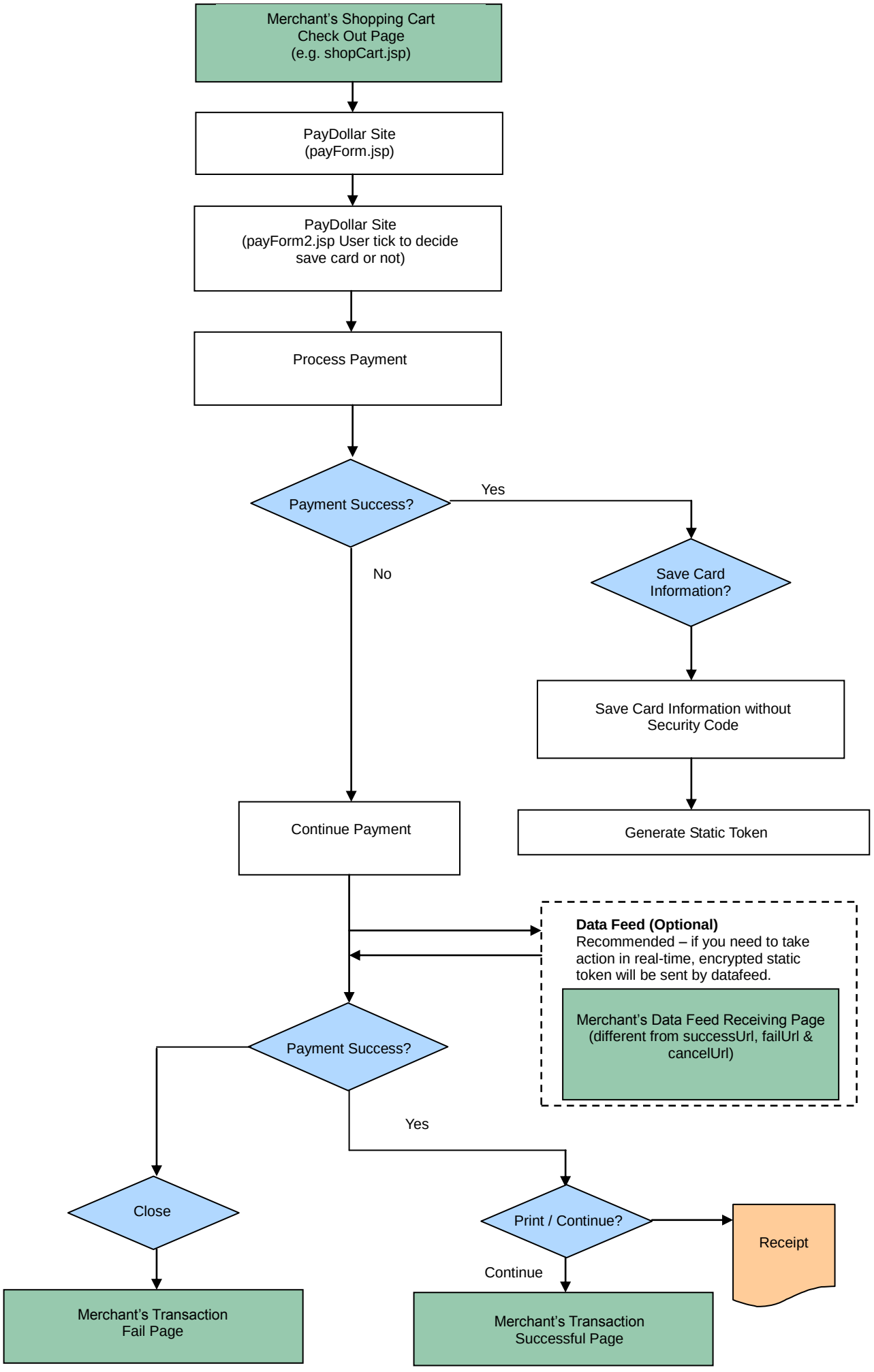
The compatible version of the software code is as follow:

Software Code	Version
HTML	4.0
Javascript	1.3

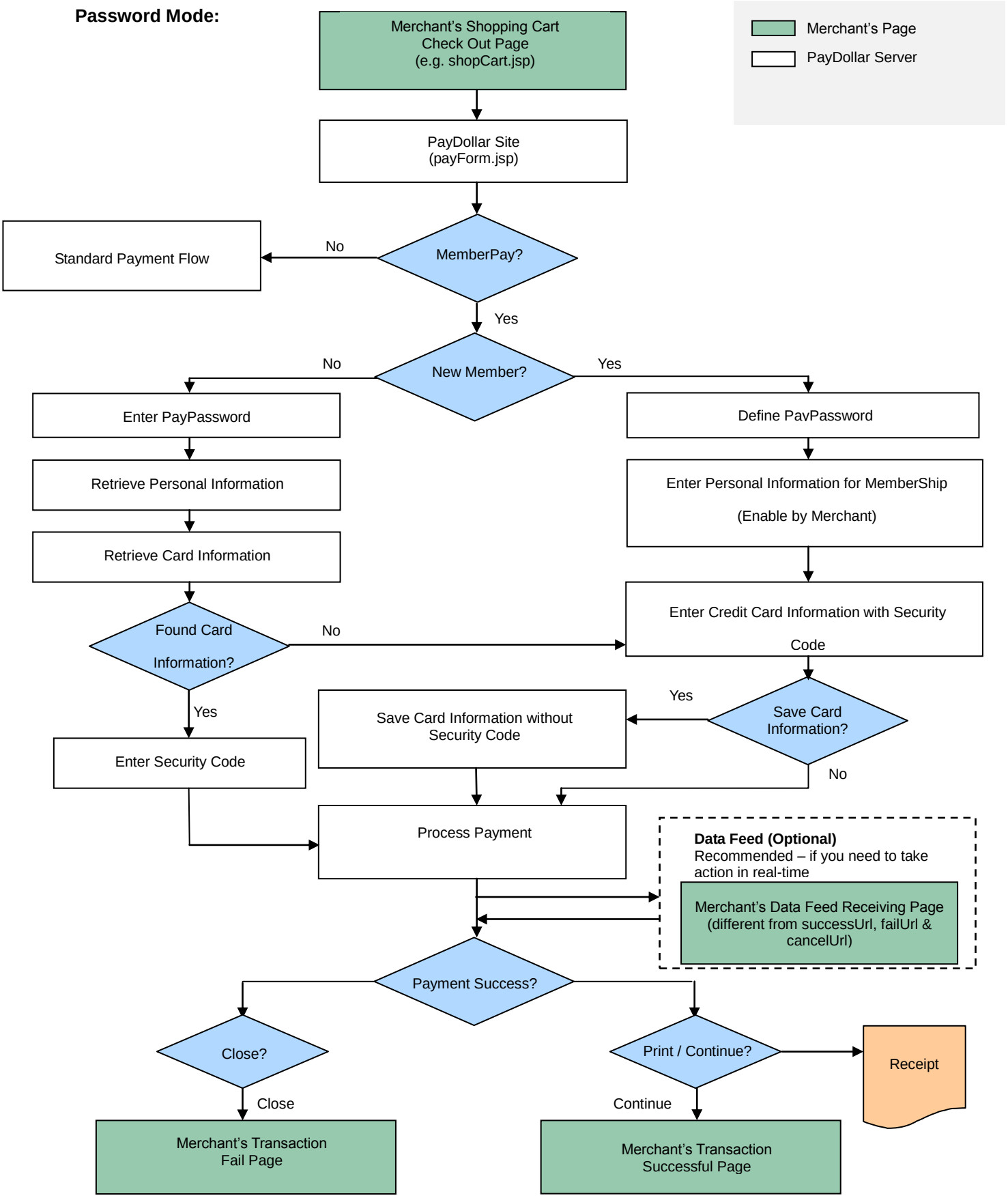
The version compatibility of the software code with popular browser software is as follow:

Browser	Version
Microsoft Internet Explorer	5.0 or above
Mozilla Firefox	3.0 or above
Google Chrome	5.0 or above
Safari	4.0 or above

Static Token Mode



Password Mode:



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 "446" – MOP "608" – PHP "764" – THB "458" – MYR "360" – IDR "410" – KRW "682" – SAR "554" – NZD "784" – AED "096" – BND "704" – VND "356" – INR 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. "E" – English "C" – Traditional Chinese "X" – Simplified Chinese "J" – Japanese "T" – Thai "F" – French "G" – German "R" – Russian "S" – Spanish "V" – Vietnamese

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) For merchants who use authorize mode, please be reminded to perform the CAPTURE action as soon as the transaction is confirmed as valid. Once captured, the customer's credit card will be debited in coming bank settlement processing. If the merchant does not capture/reverse the authorized transaction over 14 days, the credit limit will be released to the cardholder after a time period which is subjected to card issuing bank Merchant may capture/reverse the authorized transaction in the merchant administration site > Operation > Transaction Detail. Remark: Hold Payment is not available for 99BILL, ALIPAY, CHIANPAY, PAYPAL, PPS, TENPAY
payMethod	Text; (“ALL”,”PPS”,”CC”, “PAYPAL”, ”CHINAPAY”, “ALIPAY”, “TENPAY”,	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

	“99BILL)	“TENPAY” – TENPAY BY PayDollar Payment “99BILL” – 99BILL BY PayDollar Payment
memberPay_service	Text(1) (“T”,”F”)	MemberPay service indicator
memberPay_memberId	Text(50)	Member ID
memberPay_token	Text	The One-Time Token which is generated by MemberPay API (Generate MemberPay one-time token) and it can be omitted to save card again for existing member.
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.
print	Text(2) ; (“no”)	Disable the print function at payment result page.
failRetry	Text(2) ; (“no”)	Disable the retry function when the transaction is rejected
memberPay_firstName	Text(50)	Member first name
memberPay_lastName	Text(50)	Member last name
memberPay_email	Text(100)	Member email
memberPay_telephoneHome	Text(20)	Member home number
memberPay_telephoneMobile	Text(20)	Member mobile number
memberPay_telephoneOffice	Text(20)	Member office number

memberPay_billingAddress1	Text(100)	Member billing address
memberPay_billingAddress2	Text(100)	Member billing address
memberPay_billingAddress3	Text(100)	Member billing address
memberPay_billingAddress4	Text(100)	Member billing address
memberPay_shippingAddress1	Text(100)	Member shipping address
memberPay_shippingAddress2	Text(100)	Member shipping address
memberPay_shippingAddress3	Text(100)	Member shipping address
memberPay_shippingAddress4	Text(100)	Member shipping address
expDateUpdate	Text (1) ("T","F")	Only for memberPay. Flag to control whether allow user to update their expiry date of saved cards. *Enable to approved merchant only.
saveCardControl	Number	Only for memberPay. Flag to control the checkbox for save card option. There are 5 modes: 0: default setting (checked box and no alert when uncheck) 1:checked box and force enable (without alert when uncheck) 2:unchecked box and force enable (without alert when uncheck) 3:checked box and force enable (with alert when uncheck) *Enable to approved merchant only. 4:hidden check box and force enable *Enable to approved merchant only.
expDateCheck	Number	Value to control the expiry date checking. Eg. If "1" is submitted, the card expiry date must not be expired in one month. Only integer is allowed and minimum value is "1" and maximum value is "192". *Enable to approved merchant only.
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/b2cDemo/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="memberPay_service" value="T">
<input type="hidden" name="memberPay_memberId" value="20111027001">
<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:

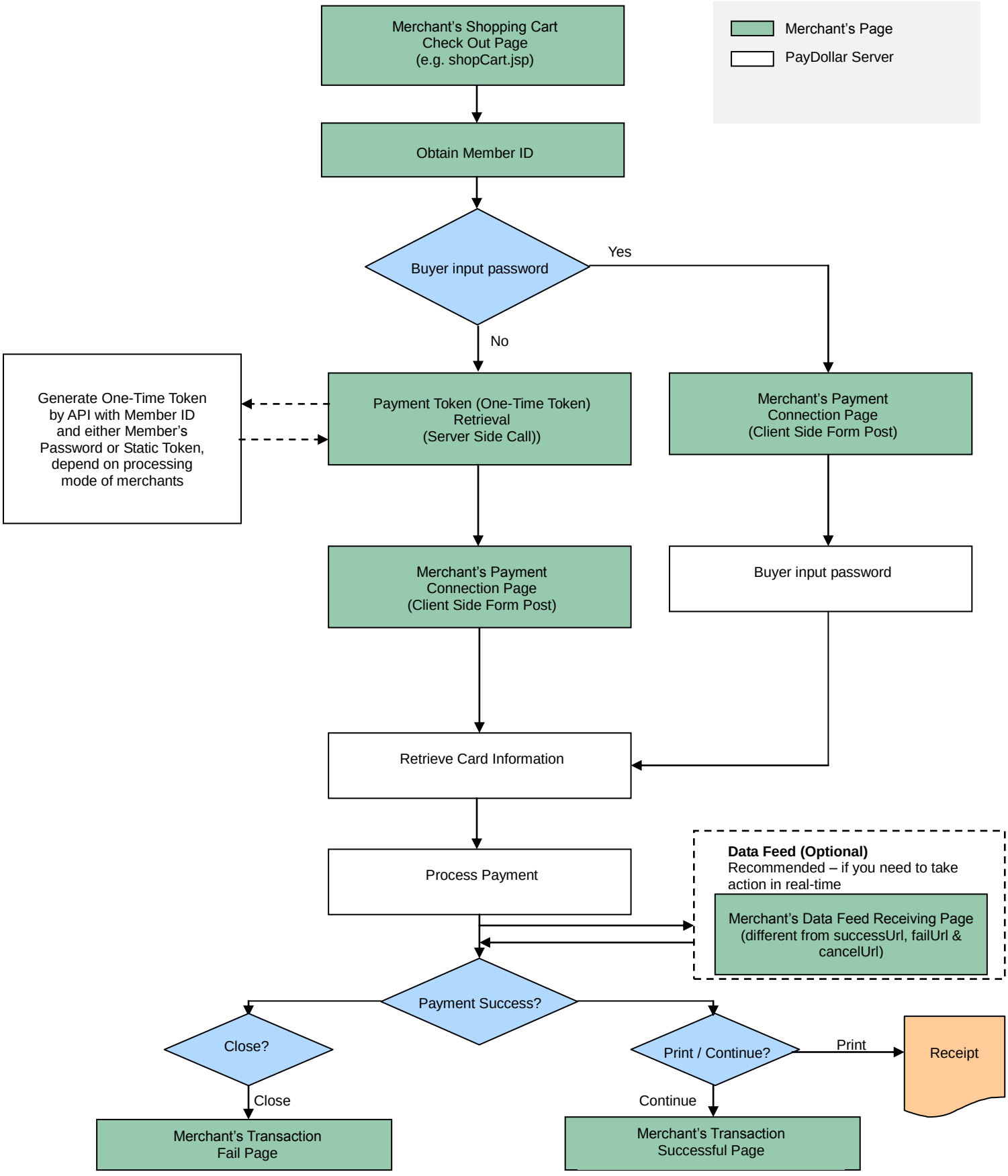
---

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

1.2 Direct Client Side Connection

Integration Procedures

For Direct Client Side Integration, merchant required to use the MemberShip API to create the membership with memberID first. Afterward, you need to post the required parameters to our payment page URL with the one-time token 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 "446" – MOP "608" – PHP "764" – THB "458" – MYR "360" – IDR "410" – KRW "682" – SAR "554" – NZD "784" – AED "096" – BND "704" – VND "356" – INR
lang	Text (1)	The language of the payment page : "E" – English "C" – Traditional Chinese "X" – Simplified Chinese "J" – Japanese "T" – Thai "F" – French "G" – German "R" – Russian "S" – Spanish "V" – Vietnamese
merchantId	Number	The merchant ID we provide to you
securityCode	Text (4)	Credit Card Verification Code - VISA: CVV2 (3-digit) - MasterCard: CVC2 (3-digit) - JCB: CAV2 (3-digit) - American Express: 4DBC (4-digit)
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) For merchants who use authorize mode, please be reminded to perform the CAPTURE action as soon as the transaction is confirmed as valid. Once captured, the customer's credit card will be debited in coming bank settlement processing. If the merchant does not capture/reverse the authorized transaction over 14 days, the credit limit will be released to the cardholder after a time period which is subjected to card issuing bank Merchant may capture/reverse the authorized transaction in the merchant administration site > Operation > Transaction Detail. Remark: Hold Payment is not available for 99BILL, ALIPAY, CHIANPAY, PAYPAL, PPS, TENPAY
memberPay_service	Text(1) ("T","F")	MemberPay service indicator
memberPay_token	Text	The One-Time Token which is generated by MemberPay API (Generate MemberPay one-time token)
memberPay_memberId	Text(50)	Member Id
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, highlighted 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="securityCode" value="123" >
<input type="hidden" name="pMethod" value="VISA" >
<input type="hidden" name="memberPay_service" value="T" >
<input type="hidden" name="memberPay_token"
value="YF8TM7JgdRuxQXI3OyTZYb0kbgHb1oyq1llqtiysCL1q9GqpinTZq5K1jqKfEVey9lHgLf0rgDtP
EHo3RDJP3w==" >
<input type="hidden" name="memberPay_memberId" value="member01" >
<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>
...
```

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:

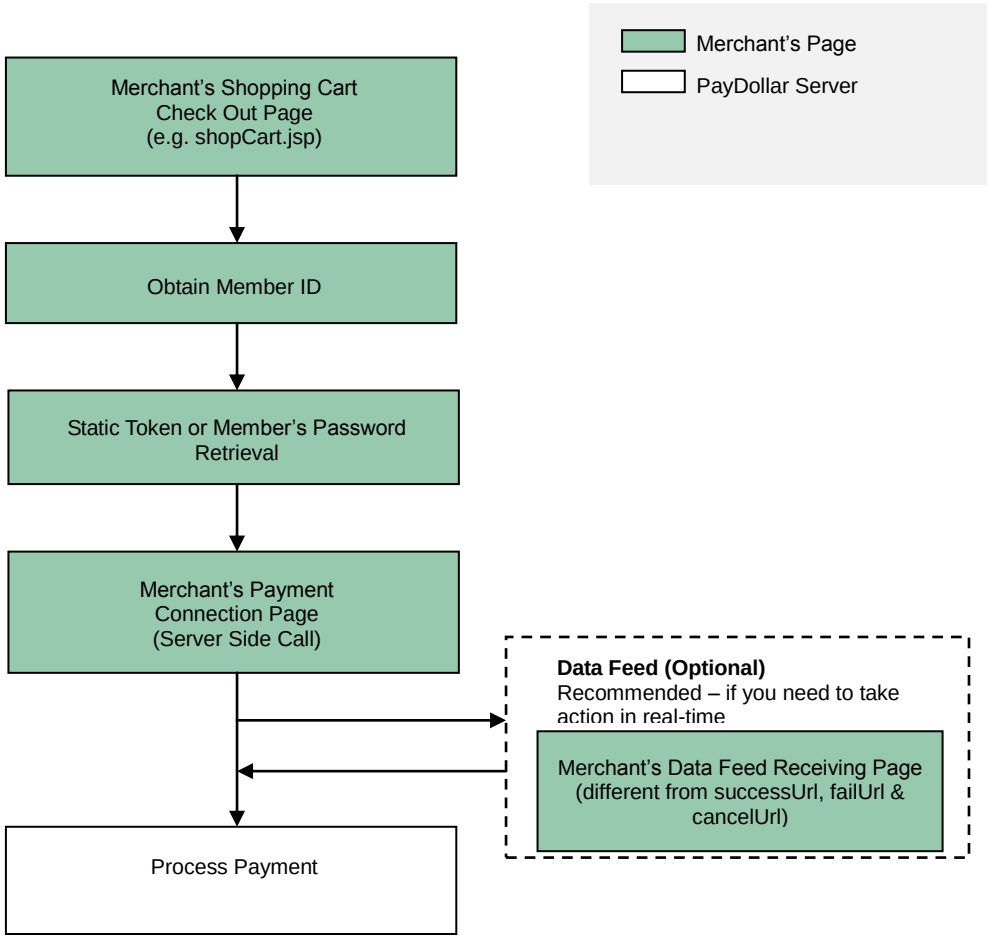
---

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

1.3 Server Side Direct Connection

Integration Procedures

For Server Side Integration, merchant required to use the MemberShip API to create the membership with memberID first. Afterward, 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

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	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 "446" – MOP "608" – PHP "764" – THB "458" – MYR "360" – IDR "410" – KRW "682" – SAR "554" – NZD "784" – AED "096" – BND "704" – VND "356" – INR
lang	Text (1)	The language of the payment page : "E" - English
securityCode	Text (4)	Credit Card Verification Code - VISA: CVV2 (3-digit) - MasterCard: CVC2 (3-digit) - JCB: CAV2 (3-digit) - American Express: 4DBC (4-digit)
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) For merchants who use authorize mode, please be reminded to perform the CAPTURE action as soon as the transaction is confirmed as valid. Once captured, the customer's credit card will be debited in coming bank settlement processing. If the merchant does not capture/reverse the authorized transaction over 14 days,

		the credit limit will be released to the cardholder after a time period which is subjected to card issuing bank Merchant may capture/reverse the authorized transaction in the merchant administration site > Operation > Transaction Detail. Remark: Hold Payment is not available for 99BILL, ALIPAY, CHIANPAY, PAYPAL, PPS, TENPAY, MEPS
memberPay_service	Text(1) ("T","F")	MemberPay service indicator
memberPay_token	Text	The One-Time Token which is generated by MemberPay API (Generate MemberPay one-time token)
memberPay_memberId	Text(50)	Member Id
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.

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&lang=E&securityCode=
123&payType=N&remark=test&memberPay_service=T&memberPay_token=YF8TM7JgdRuxQXI3OyTZY
b0kbgHb1oyq1llqtiysCL1q9GqpinTZq5K1jqKfEVey9lHgLf0rgDtPEHo3RDJP3w==&memberPay_membe
rId=member01";
// 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

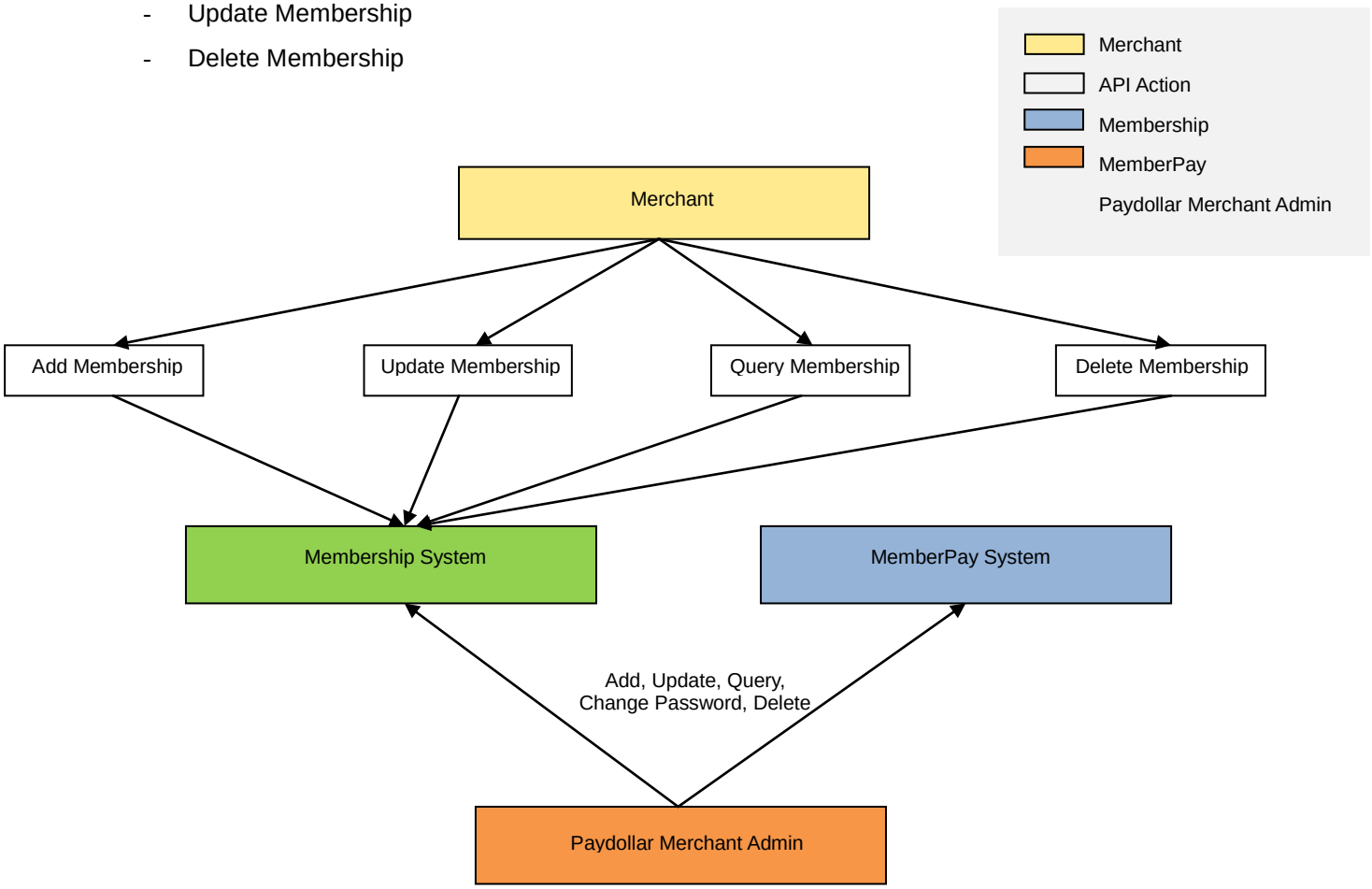
---

2 Functions of Membership API

2.1 Introduction of API functions

There are totally five functions provided:-

- Query Membership
- Add Membership
- Update Membership
- Delete Membership



To connect to our system, you need to post the required parameters by HTML form posting to our Member 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/MembershipApi.jsp

- URL of Production Platform:

  https://www.paydollar.com/b2c2/eng/merchant/api/MembershipApi.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.

2.2 Query Membership

The aim of this function is to query membership information.

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	“Query”	The action type
	memberId	Text (50)		Merchant member Id
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status
	membergroupid	Text		The member group category
	memberid	Text		The member ID of merchant provide
	firstname	Text		The first name of the member
	lastname	Text		The last name of the member
	email	Text		The email address of the member
	telephonehome	Text		The member home telephone number
	telephonemobile	Text		The member mobile telephone number
	telephoneoffice	Text		The member office telephone number
	shippingaddress1	Text		Shipping Address (part 1) of the member
	shippingaddress2	Text		Shipping Address (part 2) of the member
	shippingaddress3	Text		Shipping Address (part 3) of the member
	shippingaddress4	Text		Shipping Address (part 4) of the member
	billingaddress1	Text		Billing Address (part 1) of the

				member
	billingaddress2	Text		Billing Address (part 2) of the member
	billingaddress3	Text		Billing Address (part 3) of the member
	billingaddress4	Text		Billing Address (part 4) of the member
	status	Text		Member Status - A – Active, I – Inactive - D – Disable

All the return parameters will be in XML format

```
<membershipresponse>
    <action>Query</action>
    <responsestatus>
         <responsecode>0</responsecode>
         <responsemessage>OK</responsemessage>
    </responsestatus>
    <response>
        <membergroupid>0</membergroupid>
        <memberid>X123456</memberid>
        <firstname>Tai Ming</firstname>
         <lastname>Chan</lastname>
         <email>chan.tai.ming@abccompany.com</email>
         <telephonehome>852241234567</telephonehome>
         <telephonemobile>852241234567</telephonemoibile>
         <telephoneoffice>852241234567</telephoneoffice>
         <shippingaddress1>Rm 1293, 12/F, Block 5</shippingaddress1>
         <shippingaddress2>ABC Road</shippingaddress2>
         <shippingaddress3>Central</shippingaddress3>
         <shippingaddress4>Hong Kong</shippingaddress4>
         <billingaddress1>Rm 1293, 12/F, Block 5</billingaddress1>
         <billingaddress2>ABC Road</billingaddress2>
         <billingaddress3>Central</billingaddress3>
         <billingaddress4>Hong Kong</billingaddress4>
         <status>A</status>
     </response>
</membershipresponse>
```

2.3 Add Membership

The aim of this function is to add a membership detail.

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"Add"	The action type
	memberId	Text (50)		Merchant member Id
	firstName	Text (50)		The first name of the member
	lastName	Text (50)		The last name of the member
	memberGroupId	Text (30)		The member group category
	status	Text (1)		Member Status - A – Active, I – Inactive D – Disable
	email	Text (100)		The email address of the member
	telephoneHome	Text (20)		The member home telephone number
	telephoneMobile	Text (20)		The member mobile telephone number
	telephoneOffice	Text (20)		The member office telephone number
	shippingAddress1	Text (100)		Shipping Address (part 1) of the member
	shippingAddress2	Text (100)		Shipping Address (part 2) of the member
	shippingAddress3	Text (100)		Shipping Address (part 3) of the member
	shippingAddress4	Text (100)		Shipping Address (part 4) of the member
	billingAddress1	Text (100)		Billing Address (part 1) of the member
	billingAddress2	Text (100)		Billing Address (part 2) of the member

	billingAddress3	Text (100)		Billing Address (part 3) of the member
	billingAddress4	Text (100)		Billing Address (part 4) of the member
Optional parameter for member account				
	replace	Text(1)	{"T",'F"}	Update the information if the member exists , otherwise create new member T – enable the feature F – disable the feature [Default]
	payPassword	Text (50)		Only applied for "Password Mode". Member payment password
	account	Text (19)		Credit card number, account Id will be "1" which indicates first account of the member e.g. 4918914107195005
	expYear	Text (4)		Credit card expiry year e.g. 2015
	expMonth	Text (2)		Credit card expiry month e.g. 07
	holderName	Text (50)		Card holder Name e.g. Chan Tai Ming
	acctStatus	Text (1)		Member Account Status (if not specify, it will follow value of "Member Status") - A – Active, I – Inactive - D – Disable
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status
	statictoken	Text		Encrypted Static Token for merchant using "Static Token Mode"

```
All the return parameters will be in XML format
<membershipresponse>
    <action>Add</action>
```

```
    <responsestatus>
         <responsecode>0</responsecode>
         <responsemessage>OK</responsemessage>
    </responsestatus>
    <response>
     <statictoken>
f8qBzKBBnz6EQk1bFQF6y0cX2NTpL+WTH0Woqd73i1ExX6kRrLUaPKVvMF+dD4Fs</statictok
en>
    </response>
</membershipresponse>
```

2.4 Update Membership

The aim of this function is to update a membership

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"Update"	The action type
	memberId	Text (50)		Merchant member Id
	firstName	Text (50)		The first name of the member
	lastName	Text (50)		The last name of the member
	membergroupId	Text (30)		The member group category
	email	Text (100)		The email address of the member
	telephoneHome	Text (20)		The member home telephone number
	telephoneMobile	Text (20)		The member mobile telephone number
	telephoneOffice	Text (20)		The member office telephone number
	shippingAddress1	Text (100)		Shipping Address (part 1) of the member
	shippingAddress2	Text (100)		Shipping Address (part 2) of the member
	shippingAddress3	Text (100)		Shipping Address (part 3) of the member
	shippingAddress4	Text (100)		Shipping Address (part 4) of the member
	billingAddress1	Text (100)		Billing Address (part 1) of the member
	billingAddress2	Text (100)		Billing Address (part 2) of the member
	billingAddress3	Text (100)		Billing Address (part 3) of the member
	billingAddress4	Text (100)		Billing Address (part 4) of the

				member
	status	Text (1)		Member Status - A – Active, I – Inactive - D – Disable
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status

All the return parameters will be in XML format

```
<membershipresponse>
   <action>Update</action>
   <responsestatus>
        <responsecode>0</responsecode>
        <responsemessage>OK</responsemessage>
   </responsestatus>
   <response/>
</membershipresponse>
```

2.5 Delete Membership

The aim of this function is to delete a membership

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"Delete"	The action type
	memberId	Text (50)		Merchant member Id
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status

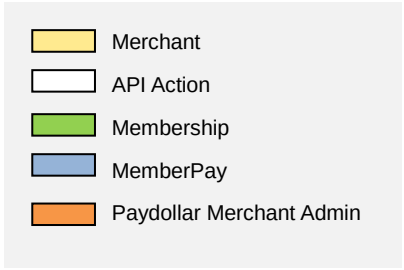
```
All the return parameters will be in XML format
<membershipresponse>
    <action>Delete</action>
    <responsestatus>
         <responsecode>0</responsecode>
         <responsemessage>OK</responsemessage>
    </responsestatus>
    <response/>
</membershipresponse>
```

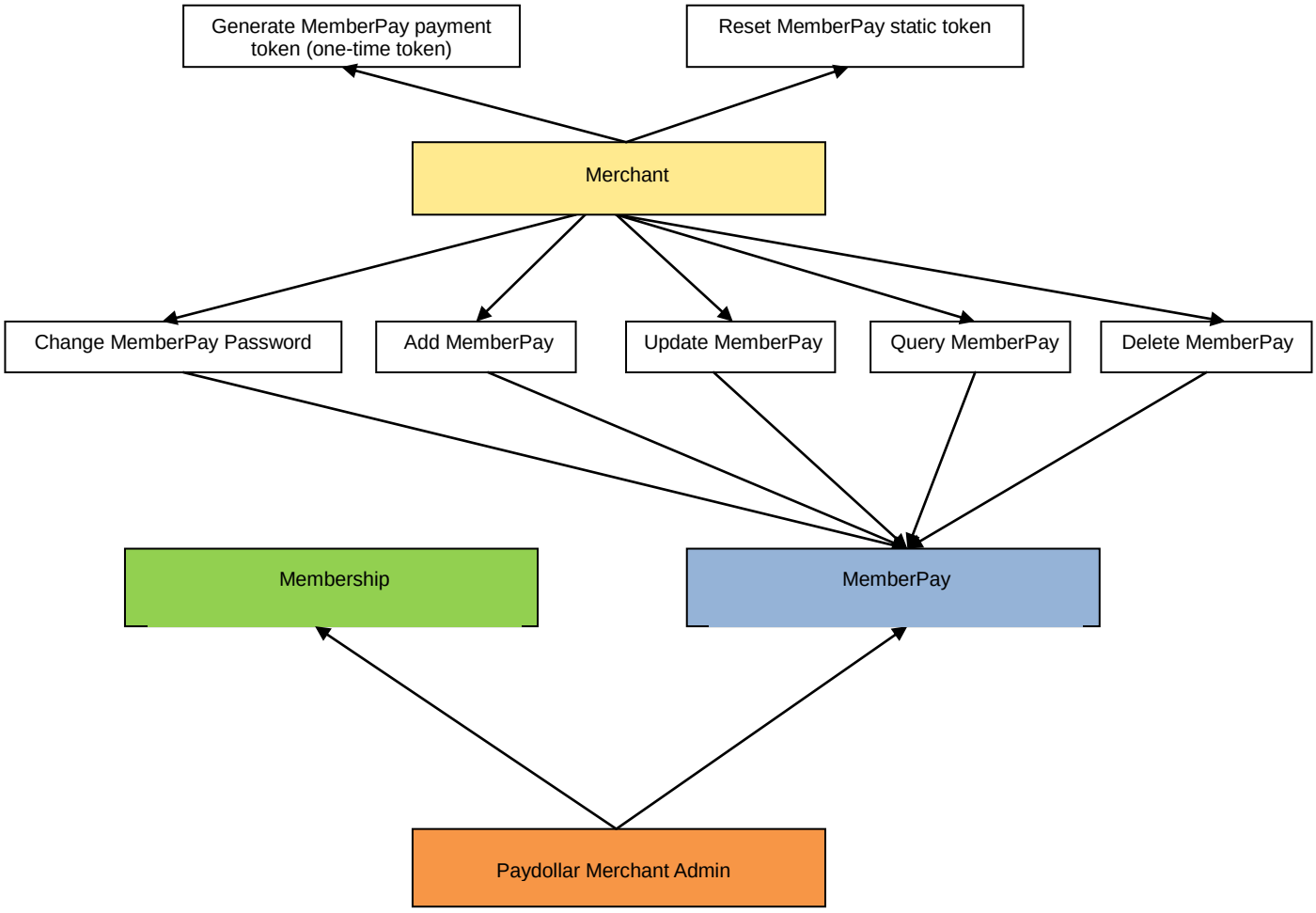
3 Functions of MemberPay API

3.1 Introduction of API functions

There are totally five functions provided:-

- Query MemberPay Account
- Add MemberPay Account
- Update MemberPay Account
- Delete MemberPay Account
- Change MemberPay Account Password
- Generate MemberPay one-time token
- Reset MemberPay static token





To connect to our system, you need to post the required parameters by HTML form posting to our MemberPay 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/MemberPayApi.jsp

- URL of Production Platform:
  https://www.paydollar.com/b2c2/eng/merchant/api/MemberPayApi.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.

3.2 Query MemberPay Account

The aim of this function is to query MemberPay account information.

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"QueryAccount"	The action type
	memberId	Text (50)		Merchant member Id
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status
	accountid	Number		Credit card order
	accounttype	Text		Credit card Type e.g. Visa
	account	Text		Credit card number e.g. 4918********5005
	expyear	Text		Credit card expiry year e.g. 2015
	expmonth	Text		Credit card expiry month e.g. 07
	holdername	Text		Card holder Name e.g. Chan Tai Ming
	statictoken	Text		Encrypted Static token for merchant using "Static Token Mode"
	status	Text		Account Status A – Active, I – Inactive D – Disable
	panFull	Text		Encrypted Card number with AES256 and encoded by Base64

```
All the return parameters will be in XML format
<memberpayresponse>
```

```
    <action>QueryAccount</action>
    <responsestatus>
         <responsecode>0</responsecode>
         <responsemessage>OK</responsemessage>
    </responsestatus>
    <response>
        <accounts>
            <accountid>1</accountid>
            <accounttype>Visa</accounttype>
            <account>491891******5005</account>
            <expyear>2015</expyear>
            <expmonth>05</expmonth>
            <holdername>test card</holdername>
<statictoken>TABA/ewnEobqVcKqyLDhh3qkug7qbTcupxG7VGzLliBy31mKPjwTXlFMPElhcS
nW</statictoken>(Only display in Static Token Mode)
             <status>A</status>
          <panFull>rN3yZg6OI0Ni7KNiu0wZiE818ScK1xyFai7U5I569rE=</panFull>
             </accounts>
        </response>
     </memberpayresponse>
```

3.3 Add MemberPay Account

The aim of this function is to add a MemberPay account

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	“Add”	The action type
	memberId	Text (50)		Merchant member Id
	account	Text (19)		Credit card number e.g. 4918914107195005
	expMonth	Text (2)		Credit card expiry month e.g. 07
	expYear	Text (4)		Credit card expiry year e.g. 2015
	holderName	Text (50)		Card holder Name e.g. Chan Tai Ming
	status	Text (1)		Account Status A – Active, I – Inactive D – Disable
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status
	accountid	Number		Credit card order
	accounttype	Text		Credit card Type e.g. Visa
	account	Text		Credit card number e.g. 491891******5005
	expyear	Text		Credit card expiry year e.g. 2015
	expmonth	Text		Credit card expiry month e.g. 07
	holdername	Text		Card holder Name e.g. Chan Tai Ming

	statictoken	Text		Encrypted Static token for merchant using “Static Token Mode”
	status	Text		Account Status A – Active, I – Inactive D – Disable

All the return parameters will be in XML format

```
<memberpayresponse>
   <action>Add</action>
   <responsestatus>
        <responsecode>0</responsecode>
        <responsemessage>OK</responsemessage>
   </responsestatus>
   <response>
      <accountid>2</accountid>
      <accounttype>MasterCard</accounttype>
      <account>542288******0007</account>
      <expyear>2015</expyear>
      <expmonth>05</expmonth>
      <holdername>test card</holdername>
    <statictoken>TABA/ewnEobqVcKqyLDhh3qkug7qbTcupxG7VGzLliBy31mKPjwTXlFMPElhcSnW</statictoken>(Only display in Static Token Mode)
      <status>A</status>
   </response>
</memberpayresponse>
```

3.4 Update MemberPay Account

The aim of this function is to update a MemberPay account

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"Update"	The action type
	memberId	Text (50)		Merchant member Id
	accountId	Number		Credit card order
	account	Text (19)		Credit card number e.g. 4918914107195005
	expMonth	Text (2)		Credit card expiry month e.g. 07
	expYear	Text (4)		Credit card expiry year e.g. 2015
	holderName	Text (50)		Card holder Name e.g. Chan Tai Ming
	status	Text (1)		Account Status A – Active, I – Inactive D – Disable
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status
	accountid	Number		Credit card order
	accounttype	Text		Credit card Type e.g. Visa
	account	Text		Credit card number e.g. 491891******5005
	expyear	Text		Credit card expiry year e.g. 2015
	expmonth	Text		Credit card expiry month e.g. 07
	holdername	Text		Card holder Name

				e.g. Chan Tai Ming
	statictoken	Text		Encrypted Static token for merchant using “Static Token Mode”
	status	Text		Account Status A – Active, I – Inactive D – Disable

All the return parameters will be in XML format

```
</memberpayresponse>
        <memberpayresponse>
          <action>Update</action>
        <responsestatus>
          <responsecode>0</responsecode>
          <responsemessage>OK</responsemessage>
          </responsestatus>
        <response>
          <accountid>1</accountid>
          <accounttype>Master</accounttype>
          <account>542288********700007</account>
          <expyear>2015</expyear>
          <expmonth>07</expmonth>
          <holdername>ap test</holdername>
          <statictoken>pyLOlgvSvClJD2+gaikeNA6bsk7sHn+WiU+EgV4D0QctD4P
lgCMKVjkA+Ec2vSrC</statictoken>(Only display in Static Token Mode)
          <status>A</status>
          </response>
          </memberpayresponse>
```

3.5 Delete MemberPay Account

The aim of this function is to delete a MemberPay account

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"Delete"	The action type
	memberId	Text (50)		Merchant member Id
	accountId	Number		Credit card order If omitted, system will delete the first active card.
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status

```
All the return parameters will be in XML format
<memberpayresponse>
    <action>Delete</action>
    <responsestatus>
        <responsecode>0</responsecode>
        <responsemessage>OK</responsemessage>
    </responsestatus>
    <response/>
</memberpayresponse>
```

3.6 Change MemberPay Account Password (Password Mode only)

The aim of this function is to change a MemberPay account password

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"ChangePassword"	The action type
	memberId	Text (50)		Merchant member Id
	newPayPassword	Text (50)		Member new payment password
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status

```
All the return parameters will be in XML format
<memberpayresponse>
    <action>ChangePassword</action>
    <responsestatus>
        <responsecode>0</responsecode>
        <responsemessage>OK</responsemessage>
    </responsestatus>
    <response/>
</memberpayresponse>
```

3.7 Generate MemberPay one-time token

The aim of this function is to generate MemberPay one-time token.

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"GenerateToken"	The action type
	memberId	Text (50)		Merchant member Id
	orderRef	Text (35)		
	currCode	Text (3)		
	amount	Number (12,2)		
	payPassword **(For "Password Mode")**	Text		Member payment password if the merchant use "Password Mode" (Merchant must use either "Password Mode" or "Static Token Mode")
	staticToken **(Mandatory For "Static Token Mode" only)**	Text		This is decrypted static token. If the merchant use "Static Token Mode" (Merchant must use either "Password Mode" or "Static Token Mode")
	accountId **(For "Password Mode")**	Number		Member credit card order. If omitted, system will reset the static token of the first active card.
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status
	token	Text		An unique ID for MemberPay service
	generatetime	Date (YYYYMMDDHH		The token generation time (GMT)

		24MISS)		e.g. 20110501223023
	timeout	Text		Session period for the token (in Second)

All the return parameters will be in XML format

```
<memberpayresponse>
    <action>GenerateToken</action>
    <responsestatus>
        <responsecode>0</responsecode>
        <responsemessage>OK</responsemessage>
    </responsestatus>
    <response>
         <token>dxlfjKLRklefafdfa=</token>
         <generatetime>20110501223023</generatetime>
         <timeout>180</timeout>
     </response>
</memberpayresponse>
```

3.8 Reset MemberPay static token (Static Token Mode Only)

The aim of this function is to re-generate MemberPay static token and return the token in xml format.

Definition of Parameters

Input /Return	Parameters (Required Fields are in Bold typeface) *Case Sensitive	Data Type	Expected Value	Descriptions
Input	**merchantId**	Number		The merchant ID we provide
	merchantApiId	Text (20)		The login ID of merchant API
	password	Text (20)		The password of merchant API
	actionType	Text (20)	"ResetStaticToken"	The action type
	memberId	Text (50)		Merchant member Id
	accountId	Number		Member credit card order. If omitted, system will reset the static token of the first active card.
output	responsecode	Number		Response code for the API request
	responsemessage	Text		Detail message for the response status
	statictoken	Text		Encrypted Static token for merchant using "Static Token Mode"

```
All the return parameters will be in XML format
<memberpayresponse>
    <action>ResetToken</action>
    <responsestatus>
        <responsecode>0</responsecode>
        <responsemessage>OK</responsemessage>
    </responsestatus>
    <response>
         <statictoken>ux4sCQZpeeCHZJ8O1juCFWeP3zRyvlrAsBSlJ2Q5YjuDpNn6Rnubeu
     z6V/hYACAd</statictoken>
     </response>
</memberpayresponse>
```

- The End -