HIPAY

“Buy, Sell and Control With Hipay”

TABLE OF CONTENTS

1/ Introduction 5

2/ Prerequisites 5

2.1 General Requirements 5
2.2 Integrating the Hipay Merchant Kit with PHP 5
2.3 Integrating with other technologies 6
2.4 Test Environment 6

3/ List of supplied files 6

4/ Glossary of important terms 7

5/ Possible payment types 7

6/ Operating principles 8

6.1 Payments and transactions 8
6.2 Process Flow 8

7/ Elements forming a payment 9

7.1 “HIPAY_MAPI_PAYMENTPARAMS” Object 9
7.2 “HIPAY_MAPI_TAX” Object 10
7.3 “HIPAY_MAPI_AFFILIATE” Object 10
7.4 “HIPAY_MAPI_ORDER” Object 10
7.5 “HIPAY_MAPI_PRODUCT” Object 10
7.6 “HIPAY_MAPI_INSTALLMENT” Object 10
7.7 “HIPAY_MAPI_SIMPLEPAYMENT” Object 10
7.8 “HIPAY_MAPI_MULTIPLEPAYMENT” Object 11
7.9 XML static functions 11

8/ “HIPAY_MAPI_PAYMENT PARAMS” Class 11

9/ “HIPAY_MAPI_TAX” Class 14

10/ “HIPAY_MAPI_AFFILIATE” Class 14

11/ “HIPAY_MAPI_ORDER” Class 15

12/ “HIPAY_MAPI_PRODUCT” Class 16

13/ “HIPAY_MAPI_INSTALLMENT” Class 17

14/ “HIPAY_MAPI_SIMPLEPAYMENT” Class 18

15/ “HIPAY_MAPI_MULTIPLEPAYMENT” Class 19

16/ “HIPAY_MAPI_SEND_XML” Class 20

17/ “HIPAY_MAPI_COMM_XML” Class 21

18/ Examples of scripts : Initiate a payment 21

18.1 Single Payment 21

a) Payment Parameters 21

b) Taxes 22

c) Affiliates 22

d) Products (order lines) 23

e) Order object 23

f) Payment object 24

g) XML representation of this order and sending the feed to the Hipay platform 24

h) Processing the platform's response 24

18.2 Recurring payment 25

a) Payment Parameters 25

b) Taxes 26

c) Affiliates 26

d) Products (products of the first payment and products of the subsequent payments) 27

e) Order objects, one for the initial payment, the other for the subsequent payments 27

f) Payment object 28

g) XML representation of this order and sending the XML feed to the Hipay platform 28

h) Processing the platform's response (XML feed [B]) 28

19/ Examples of scripts : Receiving a notification about a payment action 29

Appendix 1 : Headers added to the XML representations 30

Appendix 2 : XML data feed [A] = Single Payment posted to Hipay 30

Appendix 3 : XML data feed [A] = Recurring Payment posted to Hipay 34

Appendix 4 : XML data feed [B] returned in case of success 39

Appendix 5 : XML data feed [B] returned in case of error 39

Appendix 6 : XML data feed [C] of some notification on the payment 39

Appendix 7 : Order and product categories 41

Appendix 8 : Category of authorized age groups 41

Appendix 9 : Proxy management and network troubleshooting 41

Contact 42

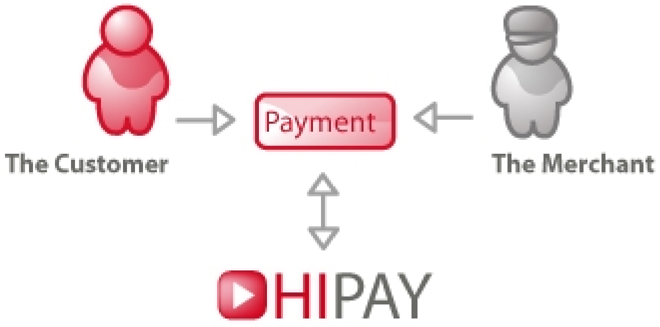
1.4 documentation version March 25st, 2011

1/ Introduction

Hipay provides merchants with a payment solution that is simple, fast and reliable for their customers. The goal of this document is to explain how to interface the merchant site with the Hipay framework.

Here is the general process of making a payment:

The merchant site transmits not only the amount to be paid by the customer, but also the breakdown of this transaction (amount per item sold, shipping charges, etc.) to the Hipay platform. The total amount of the payment must be in excess of 2€ or equivalent in other currencies. This step takes place in real time in other words, at the same time when the customer makes the payment.
The Hipay framework sends out a unique Web address. This directs the customer away from the merchant site to the unique Web address where the customer can make the payment.
The customer makes the payment.



2/ Prerequisites

2.1 General Requirements

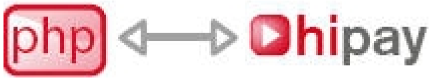
Before you can interface the Hipay framework with your site, you must first enter some information in the “Payment buttons” section of your Hipay account.
You should:
 Enter information about your merchant site so you can create a “Site ID” and a merchant password (for this go to: « Payment buttons > Add a new site »). You will be required to provide this “Site ID” and merchant password during the Merchant Kit’s installation.
If you want to distribute a part of the payments received into other accounts, you can create secondary accounts ahead of time through your “Account Manager / Account Overview”

2.2 Integrating the Hipay Merchant Kit with PHP

The Hipay Merchant API (referred to hereafter as MAPI) can only be used on a system using the PHP programming language (version 5.1 minimum) with the cURL, SimpleXML et SSL modules installed and functional.

On your web server, the “magic_quotes_gpc” option of the php.ini file must be set to “Off” in order for the merchant kit to be 100% operational.

The Merchant Kit user should have programming experience (with proficiency in object-oriented and PHP languages) as well as familiarity with current Web-related technologies.



2.3 Integrating with other technologies

The operating principle of Hipay merchant management lies in the communication (via XML data feeds) between the Hipay platform and the merchant’s platform.
However, it is possible for the user to develop his own data flow management tools and to avoid the use of MAPI entirely. In this case, the user should review this document, paying special attention to the appendices that describe the various XML feeds.
All XML requests are sent by the POST method, in secure SSL mode (https), in the “xml” parameter, to this address: https://payment.hipay.com/order/.
Hipay provides an XML schema, which can be used in coordination with the tools of your chosen technology to validate your XML feed before posting it. The schema is located at:
https://payment.hipay.com/schema/mapi.xsd

2.4 Test Environment

Hipay also makes available a “Test Environment” where you can test (offline) the integration of your system with the Hipay Framework.

The server that hosts the test platform is at this address: https://test-www.hipay.com
The payment reception for this platform URL will be https://test-payment.hipay.com/order/
This URL is included in the file ‘mapi_defs.php’ of the Merchant Kit, downloadable from the Test Environment site. Transactions entered on this site simulate the movement of funds; no actual transfer or receipt of funds will occur here.

Note that the login information for this account will not be the same as on the live platform. This is a completely independent account that will have to be set up as was the live account.

3/ List of supplied files

MAPI is provided as a series of files that provide the developer with objects that can describe the order to be paid by the customer.
“Order” means the set of data and amounts which, added together, represent the amount to be paid by the customer.

The files provided are:

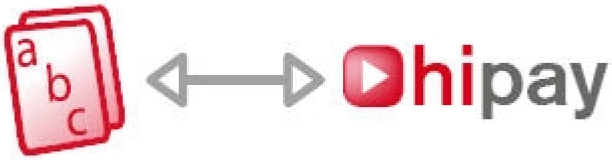
mapi_package.php *	>	File containing the components necessary for using the Kit.
mapi_defs.php *	>	Values used in the Kit.
mapi_paymentparams.php *	>	Objects that describe the general parameters of the payment.
mapi_simplepayment.php *	>	Object for single payment.
mapi_multiplepayment.php *	>	Object for subscription-type payments.
mapi_order.php *	>	Object for Order.
mapi_product.php *	>	Object for single payment form.
mapi_installment.php *	>	Object for multiple payment form.
mapi_affiliate.php *	>	Object for affiliates.
mapi_tax.php *	>	Object for taxes.
mapi_send_xml.php *	>	Object for XML data feed to the Hipay Platform.
mapi_comm_xml.php *	>	Object for for analyzing XML transfers posted by Hipay.
mapi_payment.php	>	Object for Smart Card payments.
mapi_xml.php	>	Object for internal systems conversion to XML.
mapi_utf8.php	>	Object for internal systems conversion converting to UTF-8format.
mapi_utils.php	>	Internal tool box.
mapi_exception.php	>	Object for error messages sent by the Merchant Kit in case of error.

Files followed by an asterisk (*) contain elements or objects than can be handled by the merchant. This document only describes those objects that can be handled by the merchant and those that cannot be handled will not be discussed here. The only file to include in PHP scripts using MAPI is mapi_package.php.

4/ Glossary of important terms

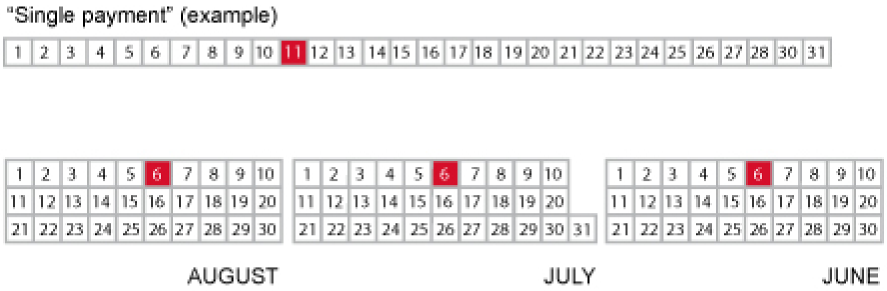
This is a list of terms you should know for better understanding of the Merchant Kit documentation:

XML: “eXtensible Markup Language” Universal data-transfer language.
MAPI : Merchant Application Programming Interface
MAPI Password: merchant password defined in the administration area of the Hipay merchant site
Merchant: Hipay member using the Hipay framework to receive client payments.
Affiliate: Hipay member who is not the merchant and who receives a part of the payment made by the customer.
Tax: A fixed amount or a percentage that will be added to the untaxed order amount.
Customer Number: The customer number is defined during the creation of your **User Space** on Hipay, and is visible in the upper right-hand corner of each page in the Hipay site’s "connected" section.
Account number: Each of the accounts belonging to a User Space has a number. The User Space “Account overview” section lists these account numbers.



5/ Possible payment types

Two types of payment are available:
“Single payment”: The customer pays the total order amount only once. Both the customer and the merchant will receive a confirmation email with the Id of the transaction.
“Recurring payment”: The customer will pay the order total at regular intervals.
In this latter case, the “recurring payment” will be linked to a subscription consisting of an initial payment (A) and subsequently a regular payment (B). Each subscription has a unique identifier.
The initial payment (A) can be granted a grace period, in which case the actual funds transfer it will be deferred to a later date, as defined by the merchant.
For each installment of the subscription, both the customer and the merchant will receive a confirmation email containing the Ids of the transaction and of the subscription itself.
The customer also will also receive reminder emails and transaction failure notices.



6/ Operating principles

6.1 Payments and transactions

A payment initiates a transaction on the Hipay platform.
A transaction consists of: an amount, a source (the customer) and a destination (the merchant).

Various operations can be applied to these:

Authorization: An operation whose purpose is to verify the solvency of the source and “hold” the transaction amount on the customer accont. The amount held in this way is not transferred to the destination (the merchant). If the source (the customer) of the transaction is a Hipay account, the authorized amount no longer appears in the customer's credit.
Cancellation: Complete or partial, the cancelled amount is released. The amount then reappears in the customer’s credit.
Capture (or "processing"): This occurs only after authorization. It enables transfer of the amount set by the authorization from the source to the transaction’s recipient.
Refund: Total or partial, a refund enables the recipient (the merchant) to reimburse the source (the customer). This can occur only after the capture has been carried out.
Rejection: Under certain circumstances, Hipay can reject the transaction. This can only occur once it has been captured.

On top of the regular emails, the merchant can receive operation notification related to a payment or to a subscription, allowing him to follow its lifecycle. These emails are copies of the standard http notifications.

6.2 Process Flow

1. The customer enters his order on the merchant site. The site uses the MAPI objects to describe the order being invoiced to the customer.

2. The merchant site transmits the order to the Hipay platform by an XML data transfer **[A].** The total amount of the order must be in excess of 2€ (or equivalent values in other currencies).

3. The Hipay platform uses an XML transfer [B] to send one of the following to the merchant a payment form, or
an error message if the XML transfer does not go through.

4. The merchant site redirects the customer to the URL of the Hipay payment interface.

5. The customer procedes with his or her payment on the Hipay interface. Once the payment operation has been completed, the customer receives an emailed confirmation of the transaction.

6. The merchant is notified of this payment by an XML data transfer **[C]** sent to a script installed on the merchant’s site. At the same time, an order information email is sent to the address that was defined in the global parameters of the payment. If this field was left empty, the email address attached to the merchants account is used.

Then, according to the type of payment, several scenarios can occur:

- Payment with immediate capture:
The transaction is authorized and captured immediately.

- Payment with deferred or manual capture:
The transaction is authorized immediately, but its capture is deferred or manually activated by the merchant in his Hipay merchant management interface.

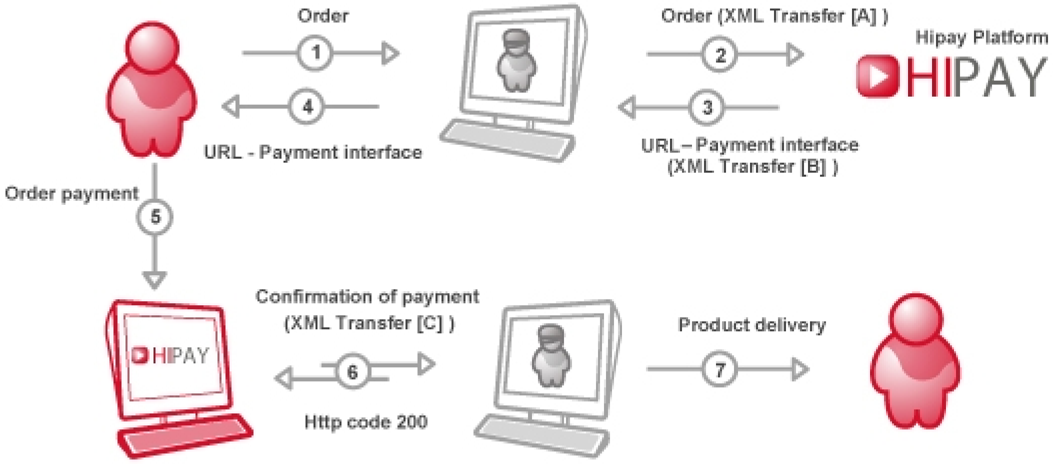
- Payment with grace period and immediate capture:
The transaction is authorized immediately, then captured when the grace period ends.

- Payment with grace period and deferred or manual capture:
The transaction is authorized immediately. The capture will be done later either automatically (programmed capture) or manually.

Throughout the process of the transaction generated by this payment, the merchant will receive notifications by XML data transfer [C] and by email. Notifications will include various actions applied to the transaction, such as cancellations and refunds.

7. The merchant delivers the content or product to his or her customer.
In the case of a recurring payment, each payment will initiate a transaction, broken down into:
An authorization
A capture
or
Simultaneous authorization and capture

Note that **only the initial payment** of a subscription can be granted a grace period.
The customer will also receive a reminder email before each charge against the account. This will remind the customer that his account should have adequate funds (in case of a payment through his Hipay account). The customer will also receive an email confirming the success or failure of the withdrawal for each recurring payment.



7/ Elements forming a payment

Regardless of whether the payment is *single* or *recurring*, an order consists of several distinct objects.

As a general rule, a payment consists of information relating to:

The Hipay platform connection.
The product(s) sold.
The taxes applied.
The affiliates.

7.1 “HIPAY_MAPI_PAYMENTPARAMS” Object

This object contains the data representing certain parameters required for the Hipay platform to process the payment:

ID number (of the Hipay account on which the website is declared) and MAPI password to connect to the platform. (« merchant password » set on the website)
Merchant accounts to where the sums paid by customers are transferred.
Website Account currency.
URL of the script on the merchant side which will receive the XML feed **[C]** informing him or her of the outcome of the payment.
URL to which the customer is directed after his or her payment, depending on the success, failure or cancellation of the transaction.

7.2 “HIPAY_MAPI_TAX” Object

This object contains the data representing a tax.

A tax is defined by its name and value; the value can be a fixed amount (in the order currency) or a percentage of all, or part, of the order amount.

7.3 “HIPAY_MAPI_AFFILIATE” Object

This object contains the data defining an affiliate.

An affiliate is defined by the customer number (User Space ID) and account number on the Hipay platform, as well as the share of the order amount that he or she will receive.
This amount is defined by a fixed value or a percentage of the order amount associated with the affiliate.

7.4 “HIPAY_MAPI_ORDER” Object

This object contains the data representing general information about the order.
These data elements are:

The amounts for shipping charges, insurance, and fixed costs associated with the order
The taxes applicable to these amounts. Taxes are represented by HIPAY_MAPI_Tax objects.
The affiliates receiving a commission on the order, represented by a set of HIPAY_MAPI_Affiliate objects.
The description and information about the order.
The merchant category of the order. (See Appendix 7)

7.5 “HIPAY_MAPI_PRODUCT” Object

This object represents an order line in the case of a “single payment”.
A “single payment” will contain this type of object, associated with a unique order.

An order line contains:
The product name and information.
The product reference.
The quantity ordered.
The product category. (See Appendix 7)
The product price, excluding taxes.
The taxes applicable to the untaxed amount, represented by HIPAY_MAPI_Tax objects.

7.6 “HIPAY_MAPI_INSTALLMENT” Object

This object represents the amount to be paid for a subscription.
The subscription will have two of these objects, one for the initial payment, and another for all the subsequent payments.
A “recurring payment” contains two objects of this type: one for the initial payment; the second to describe subsequent payments.

The data elements contained in this object are:
The untaxed amount of the payment.
The taxes applicable to this amount, represented by the objects HIPAY_MAPI_Tax.
Either the period before carrying out the payment in the case of an initial payment, or the interval between two payments for subsequent payments.

7.7 “HIPAY_MAPI_SIMPLEPAYMENT” Object

This object represents a single payment and is composed of several objects:

A HIPAY_MAPI_PaymentParams object.
A HIPAY_MAPI_Order object.
One or more HIPAY_MAPI_Product objects.

7.8 “HIPAY_MAPI_MULTIPLEPAYMENT” Object

This object represents a subscription and is composed of several objects:

A HIPAY_MAPI_PaymentParams object.
Two HIPAY_MAPI_Order objects, one for the initial payment, the other for the subsequent payments.
Two HIPAY_MAPI_Installment objects, one for the initial payment, the other for the subsequent payments.

7.9 XML static functions

HIPAY_MAPI_SEND_XML and HIPAY_MAPI_COMM_XML classes contains several functions used to send the XML [A] data feed to the Hipay platform and to retrieve the response XML [B] data feed from the platform.

8/ “HIPAY_MAPI_PAYMENT PARAMS” Class

Boolean setAccounts (int $itemAccount, int $taxAccount, int $insuranceAccount, int $fixedCostAccount, int $shippingCostAccount)
Sets the account numbers for the accounts into which the different amounts will be deposited (product sold, tax amount, insurance amount, amounts for fixed costs and shipping charges). Account numbers can be identical.

Int getItemAccount ()
Returns the account number into which the product amounts will be deposited.

Int getTaxAccount ()
Returns the account number into which the tax amount will be deposited.

Int getFixedCostAccount ()
Returns the account number into which the fixed cost amounts will be deposited.

Int getInsuranceAccount ()
Returns the account number into which the insurance amounts will be deposited.

Int getShippingCostAccount ()
Returns the account number into which the amount for shipping costs will be deposited.

Boolean setBackgroundColor (string $bg_color)
Sets the background color of the payment interface (as hexadecimal #XXXXXX).
Default color (white) is advised as image transparency remains problematic on several web browsers.
Obsolete method, prefer to set css stylesheet on website form in “payment buttons” section.

String getBackgroundColor ()
Returns the background color of the payment interface.

Boolean setCaptureDay (int $captureDay)
Defines the capture period.
The capture period corresponds to the time period, in days, between the authorization of the transaction and the transfer of the amount to the merchant’s account.
During the transaction authorization process, the amount paid by the customer is set aside and is therefore no longer available to the customer for other purchases.
The value of $captureDay can be:
HIPAY_MAPI_CAPTURE_IMMEDIATE: The capture is completed immediately.
HIPAY_MAPI_CAPTURE_MANUAL: The capture will be carried out by the merchant.
Value > 0: Number of days before capture.
The number of days before capture cannot exceed the value HIPAY_MAPI_CAPTURE_MAX_DAYS (generally 7 days, as defined by Hipay).
If manual capture is selected, the capture must occur before the HIPAY_MAPI_CAPTURE_MAX_DAYS period expires, or the transaction is cancelled and this amount is made available once again to the customer.

Int getCaptureDay ()
Returns the capture period, in days.

Boolean setCurrency (string $currency)
Defines the currency. This must be specified by its ISO-4217 code:
EUR, USD, CAD, AUD, CHF, SEK, GBP.
The currency must be the same as the website’s account.

String getCurrency ()
Returns the payment currency.

Boolean setLocale(string $defaultLang)
Sets the default language of the interface (az_AZ = language_COUNTRY)
available values : fr_FR, fr_BE, de_DE, en_GB, en_US, es_ES, nl_NL, nl_BE, pt_PT
default language is on merchant account
replace setDefaultLang which is obsolete

String getLocale()
Returns the default locale of the interface.

Boolean setEmailAck (string $email_ack)
Sets the email address for payment notification.
An email will be sent to this address after each payment, whether successful or not.
This address is generally set by merchant administrator.

String getEmailAck ()
Returns the email notification.

Boolean setIdForMerchant (string $idForMerchant)
Defines the identifier of this sale in the merchant’s system.
This alphanumeric identifier is free.

Int getIdForMerchant ()
Returns the identifier of the order for the merchant.

Boolean setLogin (string $login, string $password)
The login is the ID of the hipay website merchant account receiving the payment, and the password is the « merchant password » set within your website.
For further information, please refer to the "payment buttons > Add a new site" of the main Hipay website.

Please note: This is not the login and password used to connect to the Hipay site, but rather the login and password for connection to the gateway.
For more information, refer to the **“Payment Interface”** section on the Hipay website.

String getLogin ()
Returns the login.

String getPassword ()
Returns the password.

Boolean setLogoURL (string $logo_URL)
Sets the merchant’s logo URL, this logo will be displayed on your payment pages..
This logo, in GIF, PNG or JPG (JPEG) format must be accessible from the Internet via HTTPS protocol.
This logo must not exceed 100/100 pixels in size.

String getLogoURL ()
Returns the merchant’s logo URL.

Boolean setMedia (string $media)
Defines the payment interface type.
Currently, only the value “WEB" is allowed.

String getMedia ()
Returns the payment interface type.

Boolean setMerchantDatas (string $key, string $merchantDatas)
Sets the merchant's data in the form of key-value.
These data elements will be returned in the XML [C] data feed to the merchant.

Array getMerchantDatas ()
Returns the merchant's data in the form of a key-value table.

Boolean setMerchantSiteId (int $merchantSiteId)
Defines the identifier of the merchant's site.
This identifier corresponds to one of the sites listed in the "Merchant tool kit > Configure sites associated with the Hipay platform" section of the Hipay site.

Int getMerchantSiteId ()
Returns the identifier of the merchant's site.

Boolean setPaymentMethod (int $paymentMethod)
Defines whether the payment is *single* or *recurring*.
Possible values are:
HIPAY_MAPI_METHOD_SIMPLE: Single payment.
HIPAY_MAPI_METHOD_MULTI : Recurring payment.

Int getPaymentMethod ()
Returns the payment type (single or recurring).

Boolean setRating (string $rating)
Defines the audience rating for the product, as it relates to this payment.
The possible values are:
ALL : For all ages.
+12 : For ages 12 and over.
+16 : For ages 16 and over.
+18 : For ages 18 and over.

String getRating ()
Returns the type of public for which the product is intended.

Boolean setEmailAck (string $email_ack)
Sets the email adress to which the payment notifications will be sent.
Each phase of the payments will be accompanied by an email, whether it be a failure or a success.
This email adress should belong to the merchant.

String getEmailAck ()
Returns the ack email adress.

Boolean setURLAck (string $URL_ack)
Sets the URL to be called to notify the merchant of the results of the operations carried out on the transaction (sending of the XML feed [C]).

String getURLAck ()
Returns the URL_ack.

Boolean setURLCancel (string $URL_cancel)
Sets the URL where customers must be redirected to if the payment is cancelled.

String getURLCancel ()
Returns the URL_cancel.

Boolean setURLNok (string $URL_nok)
Sets the URL where customers must be redirected to if the payment is not authorized.

String getURLNok ()
Returns the URL_nok.

Boolean setURLOk (string $URL_ok)
Sets the URL where customers must be redirected to if the payment is authorized.

String getURLOk ()
Returns the URL_ok.

setInformations(string $text)
to display free text under the shopping cart on payment pages.

String getInformations()
get free shopping cart additional text

setIssuerAccountLogin(string $hipay_account_login)
set email of a known hipay customer account to precheck login form on payment page

Boolean getIssuerAccountLogin()
true if issuer account_ login is setted

setShopId(int $id_shop)
to replace default merchant description on payment pages. Get the shop Id on « payment buttons > shop section »

Int getShopId()
return shop identifier

Boolean check ()
Verifies that the object is correctly initialized.

9/ “HIPAY_MAPI_TAX” Class

Boolean setTaxName (string $taxName)
Sets the name of the tax.

String getTaxName ()
Returns the name of the tax.

Boolean setTaxVal (float $taxVal, [boolean $percentage = true])
Sets the value of the tax.
The value can be a fixed amount or a percentage.
In the latter case, $percentage must have the value “true”.

Float getTaxVal ()
Returns the value of the tax.

Boolean isPercentage ()
Returns “true” if the value of the tax is expressed as a percentage.

Boolean check ()
Verifies that the object is correctly initialized.

10/ “HIPAY_MAPI_AFFILIATE” Class

Boolean setCustomerId (int $customerId)
Sets the Hipay affiliate customer number.

Int getCustomerId ()
Returns the Hipay affiliate customer number.

Boolean setAccountId (int $accountId)

Sets the Hipay affiliate account number.

Int getAccountId ()
Returns the Hipay affiliate account number.

Boolean setValue (float $val, int $percentageTarget)
Sets the value of the affiliation which is either fixed amount or a percentage.
If it is a percentage, $percentageTarget represents the target, that is, on which amount(s) the affiliation amount is based.
$percentageTarget is the addition of all or part of the following values:

HIPAY_MAPI_TTARGET_TAX :
The “tax amount” is included in the percentage calculation.

HIPAY_MAPI_TTARGET_INSURANCE :
The “insurance amount” (excluding taxes) is included in the percentage calculation.

HIPAY_MAPI_TTARGET_FCOST :
The “fixed costs” amount (excluding taxes) is included in the percentage calculation.

HIPAY_MAPI_TTARGET_SHIPPING :
The “shipping cost” amount (excluding taxes) is included in the percentage calculation.

HIPAY_MAPI_TTARGET_ITEM :
The “products amount” (excluding taxes) is included in the percentage calculation.

Int getPercentageTarget ()
Returns what the percentage is applied to.

Float getValue ()
Returns the value of the affiliation.

Boolean check ()
Verifies that the object is correctly initialized.

11/ “HIPAY_MAPI_ORDER” Class

Boolean setAffiliate (array $affiliate)
Indicates the affiliates who will receive a commission for this order.
Represented by HIPAY_MAPI_Affiliate objects.

Array getAffiliate ()
Returns the list of affiliates registered for this order.

Boolean setFixedCost (float $fixedCostAmount, array $fixedCostTax)
Defines the amount of applicable fixed costs and taxes (HIPAY_MAPI_Tax objects).

Float getFixedCostAmount ()
Returns the amount of fixed costs.

Array getFixedCostTax ()
Returns the taxes applied to the fixed costs.

Boolean setInsurance (float $insuranceAmount, array $insuranceTax)
Defines the applicable amount of insurance and taxes (HIPAY_MAPI_Tax objects).

Float getInsuranceAmount ()
Returns the amount of insurance.

Array getInsuranceTax ()

Returns the taxes applicable to the insurance amount.

Boolean setShipping (float $shippingAmount, array $shippingTax)
Defines the applicable amount of shipping costs and taxes (HIPAY_MAPI_Tax objects).

Float getShippingAmount ()
Returns the shipping costs amount.

Array getShippingTax ()
Returns the taxes applicable to the shipping costs.

Boolean setOrderCategory (int $orderCategory)
Defines the category of the order.
See appendix 7 for the list of categories.

Int getOrderCategory ()
Returns the category of the order.

Boolean setOrderInfo (string $orderInfo)
Sets the information on the order.

String getOrderInfo ()
Returns the information on the order.

Boolean setOrderTitle (string $orderTitle)
Sets the title of the order:

String getOrderTitle ()
Returns the title of the order:

Boolean check ()
Verifies that the object is correctly initialized.

12/ “HIPAY_MAPI_PRODUCT” Class

Boolean setCategory (int $category)
Sets the product category.
See appendix 7 for the list of categories.

Int getCategory ()
Returns the product category.

Boolean setInfo (string $info)
Sets the information on the product.

String getInfo ()
Returns the information on the product.

Boolean setName (string $name)
Sets the product’s name.

String getName ()
Returns the product’s name.

Boolean setPrice (float $price)
Sets the unit amount, excluding taxes, for the product.

Float getPrice ()

Returns the unit amount, excluding taxes, for the product.

Boolean setQuantity (int $quantity)
Sets the product quantity.

Int getQuantity ()
Returns the product quantity.

Boolean setRef (string $ref)
Sets the merchant's preferred product reference.

String getRef ()
Returns the product reference.

Boolean setTax (array $tax)
Sets the applicable taxes for this product.
The taxes are represented by HIPAY_MAPI_Tax objects.

Array getTax ()
Returns the applicable taxes for this product.

Boolean check ()
Verifies that the object is correctly initialized.

13/ "HIPAY_MAPI_INSTALLMENT" Class

Boolean setFirst (boolean $first, string $paymentDelay)
Defines whether this is the first payment or a subsequent payment as well as the payment start time.
When $first = "true", it indicates that the object corresponds with the first payment; otherwise, the object corresponds to subsequent payments.
The meaning of the $paymentDelay value changes, depending on whether it is referring to a first or to a subsequent payment.

First payment:
In this case, the grace period must elapse before the first payment is initiated. String $paymentDelay can be expressed in hours, in days, or in months.

Subsequent payments:
This case represents the period of time between the first payment and each and every subsequent payment. Unlike the first payment, subsequent payments can only be expressed in days or months.
The value of $paymentDelay is in the form of "number/unit".
Possible units are "H" for "hour", "D" for "day" and "M" for "month".
For example: "9H" corresponds to "9 hours", "10D" corresponds to "10 days", "12M" corresponds to "12 months".
Please Note:
the unit "H" applies only to first payments; and
multiple units cannot be combined; for example, "2D5H" does not represent "2 days, 5 hours". Rather, this would be expressed as "53H" (53 hours) to achieve the same delay value.

Boolean getFirst ()
Indicates whether this is a first payment.

String getPaymentDelay ()
Returns the payment delay time. This is either:
the time that must elapse before initiating the *first* payment; or
the time between each and every *subsequent* payment.

Boolean setPrice (float $price)
Sets the payment amount, excluding taxes.

Float getPrice ()
Returns the payment amount, excluding taxes.

Boolean setTax (array $tax)
Sets the applicable taxes (HIPAY_MAPI_Tax objects) for this payment.

Array getTax ()
Returns the applicable taxes for this payment.

Boolean check ()
Verifies that the object is correctly initialized.

14/ “HIPAY_MAPI_SIMPLEPAYMENT” Class

HIPAY_MAPI_SimplePayment __construct (HIPAY_MAPI_PaymentParams $paymentParams, HIPAY_MAPI_Order $order, mixed $items)
Creates the HIPAY_MAPI_SimplePayment object from:
A HIPAY_MAPI_PaymentParams object.
A HIPAY_MAPI_Order object
A table of one or more HIPAY_MAPI_Item objects
Array getAffiliateTotalAmount ()
Returns a table of the total amount redistributed to the affiliates for each order.
Since the affiliates are shown by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
Since a HIPAY_MAPI_Order object can contain only one HIPAY_MAPI_Order object, only the first index (0) of the returned table is significant.

Float getFixedCostTaxes ()
Returns the tax amount on the fixed costs.
Since the fixed costs are shown by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
Since a HIPAY_MAPI_SimplePayment object can contain only one HIPAY_MAPI_Order object, only the first index (0) of the returned table is significant.

Float getInsuranceTaxes ()
Returns the tax amount on the insurance cost.
Since the insurance cost is shown by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
Since a HIPAY_MAPI_SimplePayment object can contain only one HIPAY_MAPI_Order object, only the first index (0) of the returned table is significant.

Float getShippingTaxes ()
Returns the tax amount on the shipping costs.
Since the shipping costs are indicated by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
Since a HIPAY_MAPI_SimplePayment object can contain only one HIPAY_MAPI_Order object, only the first index (0) of the returned table is significant.

Float getItemsTaxes ()
Returns the tax amount on the products.
Only the first index (0) of the returned table is significant.

Array getItemsTotalAmount ()
Returns the total amount of the products, excluding taxes.
Only the first index (0) of the returned table is significant.

Array getOrderTotalAmountHT ()
Returns the total amount of the order, excluding taxes.
Only the first index (0) of the returned table is significant.

Array getOrderTotalAmount ()
Returns the total amount of the order, including taxes.
Only the first index (0) of the returned table is significant.

Array getTaxesTotalAmount ()
Returns the total tax amount applied to the order.
Only the first index (0) of the returned table is significant.

Void getTotalTaxes (array &$itemsAmount, array &$shippingAmount, array &$insuranceAmount, array &$fixedCostAmount)
Returns details of the tax amounts applied to the order:
$itemAmount : Total amount of taxes applied to the products.
$shippingAmount : Total amount of taxes applied to the shipping costs.

$insuranceAmount : Total amount of taxes applied to the insurance.
$fixedCostAmount : Total amount of taxes applied to the fixed costs.
For these 4 tables, only the first index (0) is significant.

HIPAY_MAPI_PaymentParams getPaymentParams ()
Returns the parameter object (HIPAY_MAPI_PaymentParams).

Array getOrder ()
Returns the “HIPAY_MAPI_Order” object.
Only the first index (0) of the returned table is significant and will contain the object.

String getXml()
Returns the object as an XML feed in the format expected by the Hipay platform for processing.

15/ “HIPAY_MAPI_MULTIPLEPAYMENT” Class

HIPAY_MAPI_MultiplePayment _construct(HIPAY_MAPI_PaymentParams, $paymentParams,HIPAY_MAPI_Order $firstOrder,HIPAY_MAPI_Installment $firstInstallment,HIPAY_MAPI_Order $nextOrder,HIPAY_MAPI_Installment $nextInstallment)
Creates the HIPAY_MAPI_MultiplePayment object from:
- A HIPAY_MAPI_PaymentParams object.
- Two HIPAY_MAPI_Order and HIPAY_MAPI_Installment objects for the first payment.
- Two HIPAY_MAPI_Order and HIPAY_MAPI_Installment objects for subsequent payments.

Array getAffiliateTotalAmount ()
Returns the total amount redistributed to the affiliates.
Since the affiliates are shown by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
For a HIPAY_MAPI_MultiplePayment object with two HIPAY_MAPI_Order objects, only indices 0 and 1 of the returned table are significant.

Float getFixedCostTaxes ()
Returns the tax amount on the fixed costs.
Since the fixed costs are shown by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
For a HIPAY_MAPI_MultiplePayment object with two HIPAY_MAPI_Order objects, only indices 0 and 1 of the returned table are significant.

Float getInsuranceTaxes ()
Returns the tax amount on the insurance cost.
Since the insurance cost is shown by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
For a HIPAY_MAPI_MultiplePayment object with two HIPAY_MAPI_Order objects, only indices 0 and 1 of the returned table are significant.

Float getShippingTaxes ()

Returns the tax amount on the shipping costs.
Since the shipping costs are shown by the HIPAY_MAPI_Order object, the returned table indicates the total amount for each of the HIPAY_MAPI_Order objects.
For a HIPAY_MAPI_MultiplePayment object with two HIPAY_MAPI_Order objects, only indices 0 and 1 of the returned table are significant.

Float getItemsTaxes ()
Returns the tax amount on the product.The first index (0) of the returned table represents the value for the first payment, the second index (1) represents the value for the subsequent payments.

Array getItemsTotalAmount ()
Returns the total amount excluding taxes of the product.
The first index (0) of the returned table represents the value for the first payment, the second index (1) represents the value for the subsequent payments.

Array getOrderTotalAmountHT ()
Returns the total amount of the order, excluding taxes.
The first index (0) of the returned table represents the value for the first payment, the second index (1) represents the value for the subsequent payments.

Array getOrderTotalAmount ()
Returns the total amount of the order, including taxes.
The first index (0) of the returned table represents the value for the first payment, the second index (1) represents the value for the subsequent payments.

Array getTaxesTotalAmount ()
Returns the total tax amount applicable on the order.
The first index (0) of the returned table represents the value for the first payment, the second index (1) represents the value for the subsequent payments.

Void getTotalTaxes (array &$itemsAmount, array &$shippingAmount, array &$insuranceAmount,array &$fixedCostAmount)
Returns details of the tax amounts applied to the order:
$itemsAmount: Total amount of taxes applied to the products.
$shippingAmount: Total amount of taxes applied to the shipping costs.
$insuranceAmount: Total amount of taxes applied to the insurance.
$fixedCostAmount: Total amount of taxes applied to the fixed costs.
For these 4 tables, the first index (0) represents the value for the first payment, the second index (1) for the subsequent payments.

HIPAY_MAPI_PaymentParams getPaymentParams ()
Returns the parameter object (HIPAY_MAPI_PaymentParams).

Array getOrder ()
Returns the “HIPAY_MAPI_Order” object
The first index (0) returns the object concerning the first payment, the second index (1) for the subsequent payments.

String getXml()
Returns the object as an XML feed in the format expected by the Hipay platform for processing.

16/ “HIPAY_MAPI_SEND_XML” Class

Mixed sendXML($xml)
Sending an XML feed to the Hipay platform and returns the XML feed sent back by it as return.
This returned feed will be analyzed by the function analyzeResponseXML (see below).
Returns “false” in case of failure.
The URL for communication towards the Hipay platform is defined in HIPAY_GATEWAY_URL in mapi_defs.php.

IMPORTANT: while moving from the test platform the live platform, remember to update this URL.

17/ “HIPAY_MAPI_COMM_XML” Class

Boolean analyzeResponseXML(string $xml, string &$url, string &$err_msg)
Processing of the XML data feed **[B]**.
This is the feed returned by Hipay when the sendXml function is used.
Analyses the XML response sent back by the platform in response to the sendXML function:

returns the **$url** variable containing the web address where the customer is re-directed so that he can make his or her payment; or
sets err_msg to contain the error recorded in the XML feed sent by the merchant.

Boolean analyzeNotificationXML (string $xml, string &$operation, string &$status, string &$date, string &$time, string &$transid, float&$amount, string &$currency, string &$idformerchant, array &$merchantdatas, string &$emailClient, string &$subscriptionId, array &$refProduct)
Processing of the XML data feed **[C].**

Analyses the merchant payment result notification.
$xml must contain the XML data feed [C] received. It is sent by the Hipay platform using POST method, in the field named “xml” ($_POST[‘xml’]).
Returns “false” in case of error, else “true” and the different variables passed by the reference are populated with their corresponding values in the received data feed.
The operation variable indicates which type of operation has been carried out on the transaction.

The **idformerchant** and **merchantdatas** fields use the “free” values which were provided by the merchant.

IMPORTANT : These functions are not guaranteed to work if the « magic_quotes_gpc » option of your php.ini file is set to « on ».

Refer to chapter 19 and appendix 6 for further information on the returned XML feed [C].

18/ Examples of scripts : Initiate a payment

18.1 Single Payment

a) Payment Parameters

```
require_once('mapi_package.php');
$params = new HIPAY_MAPI_PaymentParams();
//The Hipay platform connection parameters. This is not the information used to connect to your Hipay
//account, but the specific login and password used to connect to the payment platform.
//The login is the ID of the hipay merchant account receiving the payment, and the password is
//the « merchant password » set within your Hipay account.
$params->setLogin('59116','MyPassword');
// The amounts will be credited to account 59118, except the taxes which will be credited to account
59119
$params->setAccounts(59118,59119);
// The payment interface will be in International French by default
$params->setLocale('en_GB');
// The interface will be the Web interface
$params->setMedia('WEB');
//The order content is intended for people at least 16 years old.
$params->setRating('+16');
// This is a single payment
$params->setPaymentMethod(HIPAY_MAPI_METHOD_SIMPLE);
// The capture take place immediately
$params->setCaptureDay(HIPAY_MAPI_CAPTURE_IMMEDIATE);
```

```
// The amounts are expressed in Euros, this has to be the same currency as the merchant’s account.
$params->setCurrency('EUR');

// The merchant-selected identifier for this order is REF6522
$params->setIdForMerchant('REF6522');
//Two data elements of type key=value are declared and will be returned to the merchant after the
payment in the notification data feed [C].
$params->setMerchantDatas('id_client','2000');
$params->setMerchantDatas('credit','10');
// This order relates to the web site which the merchant declared in the Hipay platform.
// The I.D. assigned to this website is ‘9’
$params->setMerchantSiteId(9);
// If the payment is accepted, the user will be redirected to this page
$params->setURLOk('http://www.mywebsite.com/success.html ');
// If the payment is refused, the user will be redirected to this page
$params->setUrlNok('http://www.mywebsite.com/refused.html ');
// If the user cancels the payment, he will be redirected to this page
$params->setUrlCancel('http://www.mywebsite.com/cancel.html ');
// The email address used to send the notifications, on top of the http notifications.
// cf chap 19 : RECEIVING A RESULTS NOTIFICATION ABOUT A PAYMENT ACTION
$params->setEmailAck('notification_hipay@mywebsite.com');
// The merchant’s site will be notified of the result of the payment by a call to the script
// “listen_hipay_notification.php”
// cf chap 19 : RECEIVING A RESULTS NOTIFICATION ABOUT A PAYMENT ACTION
$params->setUrlAck('http://www.mywebsite.com/listen_hipay_notification.php');

$t=$params->check();
if (!$t)
{
   echo “An error occurred while creating the paymentParams object”;
   exit;
}
```

b) Taxes

```
// Tax at 19.6%
$tax1 = new HIPAY_MAPI_Tax();
$tax1->setTaxName(‘TVA (19.6)’);
$tax1->setTaxVal(19.6,true);
$t=$tax1->check();
if (!$t)
{
   echo “An error occurred while creating a tax object”;
   exit;
}
// Fixed tax of 3.50 euros
$tax2 = new HIPAY_MAPI_Tax();
$tax2->setTaxName(‘Taxe fixe’);
$tax2->setTaxVal(3.5,false);
$t=$tax2->check();
if (!$t)
{
   echo “An error occurred while creating a tax object”;
   exit;
}
// Tax at 5.5%
$tax3 = new HIPAY_MAPI_Tax();
$tax3->setTaxName(‘TVA (5.5)’);
$tax3->setTaxVal(5.5,true);
$t=$tax3->check();
if (!$t)
{
   echo “An error occurred while creating a tax object”;
   exit;
}
```

c) Affiliates

```
// Affiliate who will receive 10% of all the items in the order
$aff1 = new HIPAY_MAPI_Affiliate();
$aff1->setCustomerId(331);
$aff1->setAccountId(59074);
$aff1->setValue(10.0,HIPAY_MAPI_TTARGET_ALL);

$t=$aff1->check();
if (!$t)
{
   echo “An error occurred while creating an affiliate object”;
   exit;
}
// Affiliate who will receive 15% of the amount of the products, insurance and delivery amounts
$aff2 = new HIPAY_MAPI_Affiliate();
$aff2->setCustomerId(332);
$aff2->setAccountId(59075);
$aff2->setValue(15.0,HIPAY_MAPI_TTARGET_ITEM | HIPAY_MAPI_TTARGET_INSURANCE |
HIPAY_MAPI_TTARGET_SHIPPING);
$t=$aff2->check();
if (!$t)
{
   echo “An error occurred while creating an affiliate object”;
   exit;
}
```

d) Products (order lines)

```
// First product: 2 copies of a book at 12.5 Euros per unit on which two taxes are applied
//(taxes $tax3 and $tax2)
$item1 = new HIPAY_MAPI_Product();
$item1->setName(‘The Fall of Hyperion’);
$item1->setInfo(‘Simmons, Dan – ISBN 0575076380’);
$item1->setquantity(2);
$item1->setRef(‘JV005’);
$item1->setCategory(5);
$item1->setPrice(12.50);
$item1->setTax(array($tax3,$tax2));
$t=$item1->check();
if (!$t)
{
   echo “An error occurred while creating a product object”;
   exit;
}
// Second product: An example of a product at 2360 Euros, on which 3 taxes are applied
//($tax1, $tax2 and $tax3)
$item2 = new HIPAY_MAPI_Product();
$item2->setName(‘PC Linux’);
$item2->setInfo(‘Computer 445’);
$item2->setquantity(1);
$item2->setRef(‘PC445’);
$item2->setCategory(2);
$item2->setPrice(2360);
$item2->setTax(array($tax1,$tax2,$tax3));
$t=$item2->check();
if (!$t)
{
   echo “An error occurred while creating a product object”;
   exit;
}
```

e) Order object

```
$order = new HIPAY_MAPI_Order();
//  Order title and information
$order->setOrderTitle(‘order on mywebsite.com’);
$order->setOrderInfo(‘best products’);
```

```
// The order category is 3 (Books)

// Refer to annex 7 to see how to find out what category your site belongs to.
$order->setOrderCategory(3);
// The shipping costs are 1.50 Euros excluding taxes, and $tax1 is applied
$order->setShipping(1.50,array($tax1));
// The insurance costs are 2 Euros excluding taxes, and $tax1 and $tax3 are applied
$order->setInsurance(2,array($tax3,$tax1));
// The fixed costs are 2.25 Euros excluding taxes, and $tax3 is applied to this amount
$order->setFixedCost(2.25,array($tax3));
// This order has two affiliates, $aff1 and $aff2
$order->setAffiliate(array($aff1,$aff2));
$t=$order->check();
if (!$t)
{
   echo “An error occurred while creating a product object”;
   exit;
}
```

f) Payment object

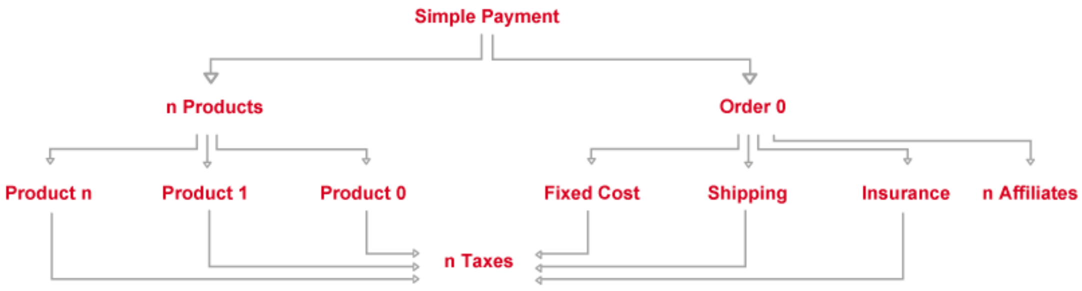
```
try {
            $payment = new HIPAY_MAPI_SimplePayment($params,$order,array($item1,$item2));
}
            catch (Exception $e) {
            echo “Error” .$e->getMessage();
}
```

g) XML representation of this order and sending the feed to the Hipay platform

```
$xmlTx=$payment->getXML();
$output=HIPAY_MAPI_SEND_XML::sendXML($xmlTx);
```

h) Processing the platform's response

```
$r=HIPAY_MAPI_COMM_XML::analyzeResponseXML($output, &$url, &$err_msg);
if ($r===true) {
            // The internet user is sent to the URL indicated by the Hipay platform
            header(‘Location: ‘.$url) ;
            // echo $url;
} else {
            // Une erreur est intervenue
            echo $err_msg;
            // $url_error = “/error.html”;
            //header('Location: '.$url_error) ;
}
```



18.2 Recurring payment

a) Payment Parameters

```
require_once('mapi_package.php');
$params = new HIPAY_MAPI_PaymentParams();
//The Hipay platform connection parameters. This is not the information used to connect to your Hipay
//account, but the specific login and password used to connect to the payment platform.
//The login is the ID of the hipay merchant account receiving the payment, and the password is
//the « merchant password » set within your Hipay account.
$params->setLogin('59116','MyMerchantPassword');
// The amounts will be credited to  account 59118, except the taxes which will be
// credited to account 59119
$params->setAccounts(59118,59119);
// The payment interface will be in international French by default
$params->setLocale('en_GB');
// The interface will be the Web interface
$params->setMedia('WEB');
//The order content is intended for people at least 16 years old.
$params->setRating('+16');
// It is a Recurring payment
$params->setPaymentMethod(HIPAY_MAPI_METHOD_MULTI);
// The capture will be immediate
$params->setCaptureDay(HIPAY_MAPI_CAPTURE_IMMEDIATE);
// The amounts are expressed in Euros, this has to be the same currency as the merchant's account.
$params->setCurrency('EUR');
// The merchant-selected identifier for this order is REF6522
$params->setIdForMerchant('REF6522');
//Two data elements of type key=value are declared and will be returned to the merchant after the
payment in the notification data feed [C].
$params->setMerchantDatas('id_client','2000');
$params->setMerchantDatas('credit','10');
// This order relates to the web site which the merchant declared in the Hipay platform.
// The I.D. assigned to this website is '9'
$params->setMerchantSiteId(9);
// If the payment is accepted, the user will be redirected to this page
$params->setURLOk('http://www.mywebsite.com/success.html ');
// If the payment is refused, the user will be redirected to this page
$params->setUrlNok('http://www.mywebsite.com/refused.html ');
// If the user cancels the payment, he will be redirected to this page
$params->setUrlCancel('http://www.mywebsite.com/cancel.html ');
// The email adress used to send the notifications, ontop of the http notifications.
// cf chap 19 : RECEIVING A RESULTS NOTIFICATION ABOUT A PAYMENT ACTION
$params->setEmailAck('notification_hipay@mywebsite.com');
// The merchant's site will be notified of the result of the payment by a call to the script
// "listen_hipay_notification.php"
// cf chap 19 : RECEIVING A RESULTS NOTIFICATION ABOUT A PAYMENT ACTION
$params->setUrlAck('http://www.mywebsite.com/listen_hipay_notification.php');

// The background color of the interface will be #FFFFFF (default color recommended)
$t=$params->setBackgroundColor('#FFFFFF');
$t=$params->check();
if (!$t)
{
   echo "An error occurred while creating the payment object";
   exit;
}
```

b) Taxes

```
// Tax at 19.6%
$tax1 = new HIPAY_MAPI_Tax();
$tax1->setTaxName(‘TVA (19.6)’);

$tax1->setTaxVal(19.6,true);
$t=$tax1->check();
if (!$t)
{
    echo “An error occurred while creating a tax object”;
    exit;
}

// Fixed tax of 3.50 euros
$tax2 = new HIPAY_MAPI_Tax();
$tax2->setTaxName(‘Taxe fixe’);
$tax2->setTaxVal(3.5,false);
$t=$tax2->check();
if (!$t)
{
    echo “An error occurred while creating a tax object”;
    exit;
}

// Tax at 5.5%
$tax3 = new HIPAY_MAPI_Tax();
$tax3->setTaxName(‘TVA (5.5)’);
$tax3->setTaxVal(5.5,true);
$t=$tax3->check();
if (!$t)
{
    echo “An error occurred while creating a tax object”;
    exit;
}
```

c) Affiliates

```
// Affiliate who will receive 10% of all the items of the order
$aff1 = new HIPAY_MAPI_Affiliate();
$aff1->setCustomerId(331);
$aff1->setAccountId(59074);
$aff1->setValue(10.0,HIPAY_MAPI_TTARGET_ALL);
$t=$aff1->check();
if (!$t)
{
    echo “An error occurred while creating an affiliate object”;
    exit;
}

// Affiliate who will receive 15% of the amount of products, insurance and delivery
$aff2 = new HIPAY_MAPI_Affiliate();
$aff2->setCustomerId(332);
$aff2->setAccountId(59075);
$aff2->setValue(15.0,HIPAY_MAPI_TTARGET_ITEM | HIPAY_MAPI_TTARGET_INSURANCE |
HIPAY_MAPI_TTARGET_SHIPPING);
$t=$aff2->check();
if (!$t)
{
    echo “An error occurred while creating an affiliate object”;
    exit;
}
```

d) Products (products of the first payment and products of the subsequent payments)

```
// First payment: The payment will be made in 1 hour, in the amount of 5 Euros, excluding taxes
// plus tax $tax1.
$ins1 = new HIPAY_MAPI_Installment();
$ins1->setPrice(5);
$ins1->setTax(array($tax1));
$ins1->setFirst(true,’1H’);
$t=$ins1->check();
if (!$t)
{
   echo “An error occurred while creating an instalment object”;
   exit;
}

// Subsequent payments: The payments will be made every 30 days in the amount of 12.5 Euros
// excluding taxes, plus tax of $tax2.0.
$ins2 = new HIPAY_MAPI_Installment();
$ins2->setPrice(12.50);
$ins2->setTax(array($tax2));
$ins2->setFirst(false,’30D’);
$t=$ins2->check();
if (!$t)
{
   echo “An error occurred while creating an instalment object”;
   exit;
}
```

e) Order objects, one for the initial payment, the other for the subsequent payments

```
// Initial order
$orderins1 = new HIPAY_MAPI_Order();
// Title and information on this payment
$orderins1->setOrderTitle(‘subscription mywebsite.com’);
$orderins1->setOrderInfo(‘1 free hour’);
// The order category is 5
// Refer to annex 7 to see how to find out what category your site belongs to.
$orderins1->setOrderCategory(5);
// the shipping costs are 1.50 Euros excluding taxes, plus tax $tax1
$orderins1->setShipping(1.50,array($tax1));
// the insurance costs are 2 Euros excluding taxes, plus taxes $tax1 and $tax3
$orderins1->setInsurance(2,array($tax3,$tax1));
// the fixed costs are 2.25 Euros excluding taxes, plus tax $tax3
$orderins1->setFixedCost(2.25,array($tax3));
// Two affiliates, $aff1 and $aff2 will receive commission for this payment.
$orderins1->setAffiliate(array($aff1,$aff2));

$t=$orderins1->check();
if (!$t)
{
   echo “An error occurred while creating an order object”;
   exit;
}

// Subsequent orders
$orderins2 = new HIPAY_MAPI_Order();
// Title and information on this payment
$orderins2->setOrderTitle(‘subscription mywebsite.com’);
$orderins2->setOrderInfo(‘only 12 euros 50 monthly !’);
// The order category is 5

// Refer to annex 7 to see how to find out what category your site belongs to.
$orderins2->setOrderCategory(5);
// The insurance costs are 5 Euros excluding taxes, plus taxes $tax1 and $tax3
```

```
$orderins2->setInsurance(5,array($tax3,$tax1));
// The fixed costs are 0.25 Euros excluding taxes, plus a tax $tax3
$orderins2->setFixedCost(0.25,array($tax3));
// An affiliate, $aff1, will receive a commission for this payment
$orderins2->setAffiliate(array($aff1,$aff2));

$t=$orderins2->check();
if (!$t)
{
   echo “An error occurred while creating an order object”;
   exit;
}
```

f) Payment object

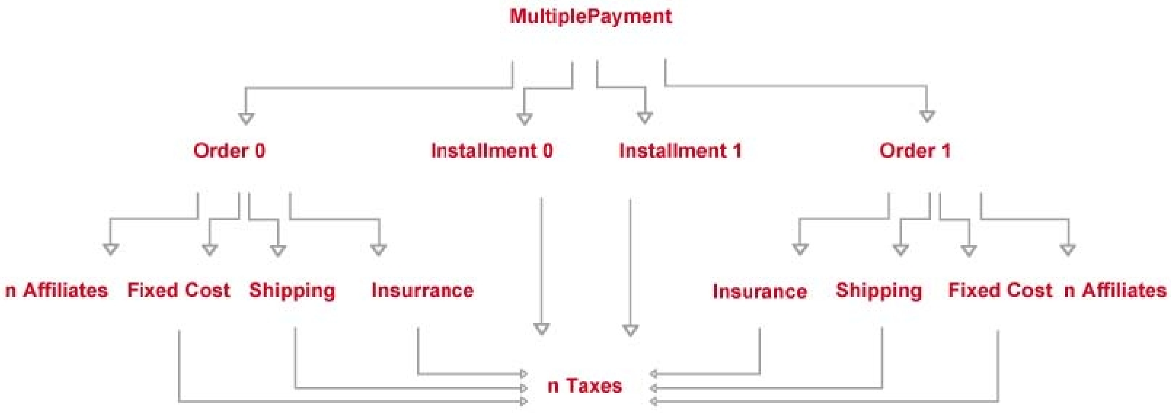
```
try {
        $payment= new HIPAY_MAPI_MultiplePayment($params,$orderins1,$ins1,$orderins2, $ins2);
} catch (Exception $e) {
        echo «Error «.$e->getMessage();
}
```

g) XML representation of this order and sending the XML feed to the Hipay platform

```
$xmlTx=$payment->getXML();
$output=HIPAY_MAPI_SEND_XML::sendXML($xmlTx);
```

h) Processing the platform's response (XML feed [B])

```
$r=HIPAY_MAPI_COMM_XML::analyzeResponseXML($output, &$url, &$err_msg);
if ($r===true) {
// The user is sent to the URL indicated by the Hipay platform
        header(‘Location: ‘.$url) ;
} else {
        // Une erreur est intervenue
        echo $err_msg;
        // $url_error = “/error.html”;
        //header('Location: '.$url_error) ;
}
```



19/ Examples of scripts : Receiving a notification about a payment action

The script below is an example script installed on the merchant site that will be called by the Hipay platform to notify the merchant of the result of a payment.
This script will be received during the XML feed **[C].**
In keeping with the previous examples, it is a script named “listen_hipay_notification.php.”

```
require_once('mapi_package.php');

// Receipt of the notification from the Hipay platform
// The XML feed [C] is sent by POST, in the field “xml”
// The function analyzeNotificationXML processes the XML feed from the Hipay platform
$r=HIPAY_MAPI_COMM_XML::analyzeNotificationXML($_POST[‘xml’], &$operation, &$status, &$date,
&$time, &$transid, &$amount, &$currency, &$idformerchant, &$merchantdatas,  &$emailClient,
&$subscriptionId, &$refProduct);

// An error occurs
if ($r===false) {
        // Error log, in a text file on the server
        file_put_contents(«result.txt», «Erreur»);
} else {
        // The feed was processed
        // Here, the merchant can update his database for orders etc.
        file_put_contents(“result.txt”,
                    "operation=$operation\n
                    status=$status\n
                    date=$date\n
                    time=$time\n
                    transaction_id=$transid\n
                    amount=$amount\n
                    currency=$currency\n
                    idformerchant=$idformerchant\n
                    merchantData=". print_r($merchantdatas,true)."\n
                    emailClient=$emailClient\n
                    subscriptionId=$subscriptionId\n
                    refProduct=".print_r($refProduct,true));
}
```

A notification is sent to the merchant for each action carried out on the payment.
These are the possible actions (represented by the value of $operation):

- **authorization** : Request for authorization in order to validate a bank card or to verify if the Hipay account of the payer has sufficient credit, with the intent to subsequently capture the transaction.
- **cancellation** : Request for total or partial cancellation.
- **refund** : Request for total or partial refund.
- **capture**: Request for capture.
- **Reject** : rejected transaction after capture

These are the possible outcomes (represented by the value of $status):

- ok: The operation ran successfully.
- nok: The operation failed.

Therefore, for a single payment with immediate capture, two notifications will be sent:
- An authorization notification
- Then, a processing notification (the processing notification taking place only if the authorization was granted).

For a recurring payment with a grace period, a pre-authorization notification will be sent when the subscription is taken, then an authorization notification and a capture notification will be sent at the end of the free period. On each recurring payment, new authorization and capture notifications will be sent. Capture notification is given immediately after the authorization notification in case of immediate processing. In case of manual or deferred processing, it is also postponed.
Other types of notifications can also be sent :

- subscription : Cancellation of a subscription (the notification « **status** » will be **cancel**).
These notifications are also sent to the ack URL.

Appendix 1 : Headers added to the XML representations

The XML feeds sent to the Hipay platform and those sent to the merchant have analogous headers in the following format:

These headers are automatically added to the XML feeds when they are sent to the Hipay platform through the function HIPAY_MAPI_SEND_XML::sendXML.

The value of the MD5 is calculated for the information following the tag </md5content> and preceding the tag </mapi>.

```
<?xml version=»1.0» encoding=»UTF-8» ?>
<mapi>
        <mapiversion>0.1</mapiversion>
        <md5content>3c4736d4d69a4af072c435665ef39408</md5content>
        [XML representation of the payment object (HIPAY_MAPI_SimplePayment ||
        HIPAY_MAPI_MultiplePayment) or of the result data in the case of a feed posted by Hipay]
</mapi>
```

Merchants who do not use MAPI should:
Enclose the XML data feed with MAPI headers.
Construct the XML representation of the payment object desired (see Appendices 2 and 3).
Post a POST request with the field "XML" set to the value of the XML feed, sent to these addresses
https://payment.hipay.com/order/ on the production platform
https://test-payment.hipay.com/order/ on the test platform.
In response, the merchant will receive an XML data feed (shown in Appendices 4 and 5).

Hipay provides an XML schema, which can be used in coordination with the tools of your chosen technology to validate your XML feed before posting it. The schema is located at:
https://payment.hipay.com/schema/mapi.xsd

Appendix 2 : XML data feed [A] = Single Payment posted to Hipay

(Reference Section18)

Parent Tag	tag	Required ?	Description
HIPAY_MAPI_SimplePayment	HIPAY_MAPI_PaymentParams	YES	Global parameters of the payment
HIPAY_MAPI_PaymentParams	login	YES	Int. ID of the website user account
HIPAY_MAPI_PaymentParams	password	YES	Char. Website Merchant Password
HIPAY_MAPI_PaymentParams	itemAccount	YES	Int. ID of the account where the non-taxed product amount will be returned
HIPAY_MAPI_PaymentParams	taxAccount	YES	Int. ID of the account where the taxed product amount will be returned
HIPAY_MAPI_PaymentParams	insuranceAccount	YES	Int. ID of the account where the non-taxed insurance cost will be returned
HIPAY_MAPI_PaymentParams	fixedCostAccount	YES	Int. ID of the account where the non-taxed fixed cost will be returned
HIPAY_MAPI_PaymentParams	shippingCostAccount	YES	Int. ID of the account where the non-taxed

			shipping cost will be returned
HIPAY_MAPI_PaymentParams	defaultLang	YES	Char. Obsolete, the language of the merchant's Hipay account is used.
HIPAY_MAPI_PaymentParams	media	YES	Char. The type of media (WEB is the only value available at the moment)
HIPAY_MAPI_PaymentParams	rating	YES	Char. Age Class authorized for processing this payment (Values are available in Appendix 8)
HIPAY_MAPI_PaymentParams	paymentMethod	YES	Int. Payment method: Simple (value=0) or multiple (value=1) here = 0
HIPAY_MAPI_PaymentParams	captureDay	YES	Int. Capture delay, or period of time, in days, separating the authorization from the transaction and the transfer from the sum on the merchant account. Possible values: -1 (manual capture) 0 (immediate capture) 0<x<7 (capture after x days)
HIPAY_MAPI_PaymentParams	currency	YES	Char. Currency of the payment indicated by its ISO-4217 code : EUR, USD, CAN, AUD, CHF, SEK, GBP
HIPAY_MAPI_PaymentParams	idForMerchant	YES	Char. Identification of the transaction merchant
HIPAY_MAPI_PaymentParams	merchantSiteId	YES	Int. Identification of the merchant site
HIPAY_MAPI_PaymentParams	merchantDatas	YES	Merchant data elements open, returned in the notification requests (XML data feed [C]) in the form <_aKey_[key]>[value]</_aKey_[key]>
HIPAY_MAPI_PaymentParams	url_ok	YES	Char. URL to which customer is redirected after a successful payment
HIPAY_MAPI_PaymentParams	url_nok	YES	Char. URL to which customer is redirected after a failed payment
HIPAY_MAPI_PaymentParams	url_cancel	YES	Char. URL to which customer is redirected after a payment that is cancelled by the customer
HIPAY_MAPI_PaymentParams	url_ack	YES	Char. URL used by Hipay to post operation notifications
HIPAY_MAPI_PaymentParams	email_ack	YES	Char. Email used by Hipay to post operation notifications
HIPAY_MAPI_PaymentParams	bg_color	YES	Char. Hexadecimal hexadecimal code of the background color of the payment pages, preceded by # (ex. #DF45U7)
HIPAY_MAPI_PaymentParams	logo_url	YES	Char. URL of image file of the merchant’s logo that will be attached to payment pages
HIPAY_MAPI_SimplePayment	order	YES	List of orders. In the single payment form, the order is unique
order	HIPAY_MAPI_Order	YES (1and only 1)	Order object
HIPAY_MAPI_Order	shippingAmount	YES	Float. Amount of fixed shipping cost
HIPAY_MAPI_Order	shippingTax	YES	Taxes applied to shipping cost
HIPAY_MAPI_Order	insuranceAmount	YES	Float. Amount of fixed insurance cost
HIPAY_MAPI_Tax	taxName	YES	Char. Tax description
HIPAY_MAPI Tax	taxVal	YES	Float. Tax value
HIPAY_MAPI_Tax	percentage	YES	Boolean. True if the value is in percentage, false if not
HIPAY_MAPI_SimplePayment	items	YES	Product list (number of products possible)
items	HIPAY_MAPI_Product	YES	Product object
HIPAY_MAPI_Product	name	YES	Char. Product name

HIPAY_MAPI_Product	info	YES	Char. Product description
HIPAY_MAPI_Product	quantity	YES	Int. Number of product units in the order
HIPAY_MAPI_Product	ref	YES	Char. Merchant reference number of the product
HIPAY_MAPI_Product	category	YES	Int. Product category (See Appendix 7)
HIPAY_MAPI_Product	price	YES	Float. Product price, excluding taxes
HIPAY_MAPI_Product	tax	YES	List of taxes applicable to the product

This representation is the one returned by the getXML function of the object $payment. More details at https://payment.hipay.com/schema/mapi.xsd

```
<HIPAY_MAPI_SimplePayment>
        <HIPAY_MAPI_PaymentParams>
                <login>59116</login>
                <password>MyMerchantPassword</password>
                <itemAccount>59118</itemAccount>
                <taxAccount>59119</taxAccount>
                <insuranceAccount>59118</insuranceAccount>
                <fixedCostAccount>59118</fixedCostAccount>
                <shippingCostAccount>59118</shippingCostAccount>
                <defaultLang>en_US</defaultLang>
                <media>WEB</media>
                <rating>+16</rating>
                <paymentMethod>0</paymentMethod>
                <captureDay>0</captureDay>
                <currency>EUR</currency>
                <idForMerchant>REF6522</idForMerchant>
                <merchantSiteId>9</merchantSiteId>
                <merchantDatas>
                        <_aKey_id_client>2000</_aKey_id_client>
                        <_aKey_credit>10</_aKey_credit>
                </merchantDatas>
                <url_ok>http://www.mywebsite.com/success.html</url_ok>
                <url_nok>http://www.mywebsite.com/refused.html</url_nok>
                <url_cancel>http://www.mywebsite.com/cancel.html</url_cancel>
                <url_ack>http://www.mywebsite.com/listen_hipay_notification.php</url_ack>
                <email_ack>hipay_notification@mywebsite .com</email_ack>
                <bg_color>#FFFFFF</bg_color>
                <logo_url>https://www.mywebsite.com/img/logo.jpg</logo_url>
        </HIPAY_MAPI_PaymentParams>
        <order>
                <HIPAY_MAPI_Order>
                        <shippingAmount>1.50</shippingAmount>
                        <shippingTax>
                                <HIPAY_MAPI_Tax>
                                        <taxName>TVA (19.6)</taxName>
                                        <taxVal>19.60</taxVal>
                                        <percentage>true</percentage>
                                </HIPAY_MAPI_Tax>
                        </shippingTax>
                        <insuranceAmount>2.00</insuranceAmount>
                        <insuranceTax>
                                < H I P A Y_MAPI_Tax>
                                        <taxName>TVA (5.5)</taxName>
                                        <taxVal>5.50</taxVal>
                                        <percentage>true</percentage>
                                </HIPAY_MAPI_Tax>
                                <HIPAY_MAPI_Tax>
                <taxName>TVA (19.6)</taxName>
                <taxVal>19.60</taxVal>
                <percentage>true</percentage>
            </HIPAY_MAPI_Tax>
        </insuranceTax>
        <fixedCostAmount>2.25</fixedCostAmount>
        <fixedCostTax>
            <HIPAY_MAPI_Tax>
                <taxName>TVA (5.5)</taxName>
                <taxVal>5.50</taxVal>
                <percentage>true</percentage>
            </HIPAY_MAPI_Tax>
        </fixedCostTax>
        <affiliate>
            <HIPAY_MAPI_Affiliate>
                <customerId>331</customerId>
                <accountId>59074</accountId>
                <val>10.00</val>
                <percentageTarget>31</percentageTarget>
            </HIPAY_MAPI_Affiliate>
            <HIPAY_MAPI_Affiliate>
                <customerId>332</customerId>
                <accountId>59075</accountId>
                <val>15.00</val>
                <percentageTarget>26</percentageTarget>
            </HIPAY_MAPI_Affiliate>
        </affiliate>
        <orderTitle>order on mywebsite.com</orderTitle>
        <orderInfo>best products</orderInfo>
        <orderCategory>3</orderCategory>
    </HIPAY_MAPI_Order>
</order>
<items>
    <HIPAY_MAPI_Product>
        <name>The Fall of  Hyperion</name>
        <info>Simmons, Dan – ISBN 0575076380</info>
        <quantity>2</quantity>
        <ref>JV005</ref>
        <category>5</category>
        <price>12.50</price>
        <tax>
            <HIPAY_MAPI_Tax>
                <taxName>TVA (5.5)</taxName>
                <taxVal>5.50</taxVal>
                <percentage>true</percentage>
            </HIPAY_MAPI_Tax>
            <HIPAY_MAPI_Tax>
                <Name>Taxe fixe</taxName>
                <taxVal>3.50</taxVal>
                <percentage>false</percentage>
            </HIPAY_MAPI_Tax>
        </tax>
    </HIPAY_MAPI_Product>
    <HIPAY_MAPI_Product>
        <name>PC Linux</name>
        <info>4Go ram</info>
        <quantity>1</quantity>
        <ref>PC445</ref>
        <category>2</category>
        <price>2360.00</price>
```

```
            <tax>
                <HIPAY_MAPI_Tax>
                    <taxName>TVA (19.6)</taxName>
                    <taxVal>19.60</taxVal>
                    <percentage>true</percentage>
                </HIPAY_MAPI_Tax>
                <HIPAY_MAPI_Tax>
                    <taxName>Taxe fixe</taxName>
                    <taxVal>3.50</taxVal>
                    <percentage>false</percentage>
                </HIPAY_MAPI_Tax>
                <HIPAY_MAPI_Tax>
                    <taxName>TVA (5.5)</taxName>
                    <taxVal>5.50</taxVal>
                    <percentage>true</percentage>
                </HIPAY_MAPI_Tax>
            </tax>
        </HIPAY_MAPI_Product>
    </items>
</HIPAY_MAPI_SimplePayment>
```

Appendix 3 : XML data feed [A] = Recurring Payment posted to Hipay

(reference section 19)

Parent tag	Tag	Required	Description
HIPAY_MAPI_MultiplePayment	HIPAY_MAPI_PaymentParams	YES	General payment parameters
HIPAY_MAPI_PaymentParams	login	YES	Int. ID of website user account
HIPAY_MAPI_PaymentParams	password	YES	Char. Website merchant password
HIPAY_MAPI_PaymentParams	itemAccount	YES	Int. ID of the account where the non-taxed product amount will be returned
HIPAY_MAPI_PaymentParams	taxAccount	YES	Int. ID of the account where the taxed product amount will be returned
HIPAY_MAPI_PaymentParams	insuranceAccount	YES	Int. ID of the account where the non-taxed insurance cost will be returned
HIPAY_MAPI_PaymentParams	fixedCostAccount	YES	Int. ID of the account where the non-taxed fixed cost will be returned
HIPAY_MAPI_PaymentParams	shippingCostAccount	YES	Int. ID of the account where the non-taxed shipping cost will be returned
HIPAY_MAPI_PaymentParams	defaultLang	YES	Char. Obsolete, the language of the merchant's Hipay account is used.
HIPAY_MAPI_PaymentParams	media	YES	Char. The type of media (WEB is the only value available at the moment)
HIPAY_MAPI_PaymentParams	rating	YES	Char. Age Class authorized for processing this payment (Values are available in Appendix 8)

HIPAY_MAPI_PaymentParams	paymentMethod	YES	Int. Payment method: Simple (value=0) or multiple (value=1) here = 1
HIPAY_MAPI_PaymentParams	captureDay	YES	Int. Capture delay, or period of time, in days, separating the authorization from the transaction and the transfer from the sum on the merchant account. Possible values: -1 (manual capture) 0 (immediate capture) 0<x<7 (capture after x days)
HIPAY_MAPI_PaymentParams	currency	YES	Char. Currency of the payment indicated by its ISO-4217 code : EUR, USD, CAN, AUD, CHF, SEK, GBP Must be the currency of the merchant account
HIPAY_MAPI_PaymentParams	idForMerchant	YES	Char. Identification of the transaction merchant
HIPAY_MAPI_PaymentParams	merchantSiteId	YES	Int. Identification of the merchant site
HIPAY_MAPI_PaymentParams	merchantDatas	YES	Merchant data elements open, returned in the notification requests (XML data feed [C]) in the form <_aKey_[key]>[value]</_aKey_[key]>
HIPAY_MAPI_PaymentParams	url_ok	YES	Char. URL to which customer is redirected after a successful payment
HIPAY_MAPI_PaymentParams	url_nok	YES	Char. URL to which customer is redirected after a failed payment
HIPAY_MAPI_PaymentParams	url_cancel	YES	Char. URL to which customer is redirected after a payment that is cancelled by the customer
HIPAY_MAPI_PaymentParams	url_ack	YES	Char. URL used by Hipay to post operation notifications
HIPAY_MAPI_PaymentParams	email_ack	YES	Char. Email used by Hipay to Hipay to post operation notifications
HIPAY_MAPI_PaymentParams	bg_color	YES	Char. Hexadecimal hexadecimal code of the background color of the payment pages, preceded by # (#FFFFFF recommended)
HIPAY_MAPI_PaymentParams	logo_url	YES	Char. URL of image file of the merchant's logo that will be attached to payment pages
HIPAY_MAPI_MultiplePayment	order	YES	List of orders. In the multiple payment form, there are 2 orders
order	HIPAY_MAPI_Order	YES	Order object
HIPAY_MAPI_Order	shippingAmount	NO	Float. Amount of fixed shipping cost
HIPAY_MAPI_Order	shippingTax	NO	Taxes applied to shipping cost
HIPAY_MAPI_Order	insuranceAmount	NO	Float. Amount of fixed insurance cost
HIPAY_MAPI_Order	insuranceTax	NO	Taxes applied to insurance cost
HIPAY_MAPI_ Order	fixedCostAmount	NO	Float. Amount of fixed cost
HIPAY_MAPI_Order	fixedCostTax	NO	Taxes applied to fixed cost

HIPAY_MAPI_Order	affiliate	NO	List of affiliates
affiliate	HIPAY_MAPI_Affiliate	NO	Affiliate object
HIPAY_MAPI_Affiliate	customerId	YES	Int. ID of the Hipay affiliate’s User Space
HIPAY_MAPI_Affiliate	accountId	YES	Int. ID of the Affiliate account to be credited
HIPAY_MAPI_Affiliate	val	YES	Float. Amount (fixed or percentage) to be returned to affliate (Ex. 10.0)

HIPAY_MAPI_Affiliate	percentageTarget	YES	Int.
			If it is a percentage, the value is equal to the sum of the target values:
			TAX = 1
			INSURANCE = 2
			FCOST = 4
			SHIPPING = 8
			ITEM = 16
			ALL = 1+2+4+8+16 = 31
			Ex. If the targets are tax and insurance, percentageTarget = 3

HIPAY_MAPI_Order	orderTitle	YES	Char. Order title
HIPAY_MAPI_Order	orderInfo	YES	Char. Description of the order
HIPAY_MAPI_Order	orderCategory	YES	Int. Order category (see Appendix 7)
ShippingTax, insuranceTax, fixedCostTax, tax	HIPAY_MAPI_Tax	NO	Tax object
HIPAY_MAPI_Tax	taxName	YES	Char. Tax description
HIPAY_MAPI_Tax	taxVal	YES	Float. Tax value
HIPAY_MAPI_Tax	percentage	YES	Boolean. True if the value is in percentage, false if not
HIPAY_MAPI_MultiplePayment	items	YES	Product list (2 and only 2 products)
items	HIPAY_MAPI_Installment	YES (2 and only 2)	Subscription product order
HIPAY_MAPI_Installment	price	YES	Float. Amount product
HIPAY_MAPI_Installment	first	YES	Boolean. Indicates whether this payment is the first or the sercond one.
HIPAY_MAPI_Installment	paymentDelay	YES	Char. Defines the duration of the grace period if carried by the first product (format: xHDM), or the gap between two sequential payments if carried by the second product (format: xDM).
HIPAY_MAPI_Installment	tax	YES	Taxes carried by the product.

This representation is the one returned by the getXML function of the object $payment. More details at https://payment.hipay.com/schema/mapi.xsd

```
    <HIPAY_MAPI_MultiplePayment>
<HIPAY_MAPI_PaymentParams>
        <login>59116</login>
        <password>MyMerchantPassword</password>
        <itemAccount>59118</itemAccount>
        <taxAccount>59119</taxAccount>
        <insuranceAccount>59118</insuranceAccount>
        <fixedCostAccount>59118</fixedCostAccount>
        <shippingCostAccount>59118</shippingCostAccount>
        <defaultLang>en_GB</defaultLang>
        <media>WEB</media>
        <rating>+16</rating>
        <paymentMethod>1</paymentMethod>
        <captureDay>0</captureDay>
        <currency>EUR</currency>
        <idForMerchant>REF6522</idForMerchant>
        <merchantSiteId>9</merchantSiteId>
        <merchantDatas>
            <_aKey_id_client>2000</_aKey_id_client>
            <_aKey_credit>10</_aKey_credit>
        </merchantDatas>
        <url_ok>http://www.mywebsite.com/success.html</url_ok>
        <url_nok>http://www.mywebsite.com/refused.html</url_nok>
        <url_cancel>http://www.mywebsite.com/cancel.html</url_cancel>
        <url_ack>http://www.mywebsite.com/listen_hipay_notification.php</url_ack>
        <email_ack>hipay_notification@mysite .com</email_ack>
        <bg_color>#FFFFFF</bg_color>
        <logo_url>https://www.mywebsite.com/img/logo.jpg</logo_url>
    </HIPAY_MAPI_PaymentParams>
    <order>
        <HIPAY_MAPI_Order>
            <shippingAmount>1.50</shippingAmount>
            <shippingTax>
                <HIPAY_MAPI_Tax>
                    <taxName>TVA (19.6)</taxName>
                    <taxVal>19.60</taxVal>
                    <percentage>true</percentage>
                </HIPAY_MAPI_Tax>
            </shippingTax>
            <insuranceAmount>2.00</insuranceAmount>
            <insuranceTax>
                <HIPAY_MAPI_Tax>
                    <taxName>TVA (5.5)</taxName>
                    <taxVal>5.50</taxVal>
                    <percentage>true</percentage>
                </HIPAY_MAPI_Tax>
                <HIPAY_MAPI_Tax>
                    <taxName>TVA (19.6)</taxName>
                    <taxVal>19.60</taxVal>
                    <percentage>true</percentage>
                </HIPAY_MAPI_Tax>
            </insuranceTax>
            <fixedCostAmount>2.25</fixedCostAmount>
            <fixedCostTax>
                <HIPAY_MAPI_Tax>
                    <taxName>TVA (5.5)</taxName>
                    <taxVal>5.50</taxVal>
                    <percentage>true</percentage>
                </HIPAY_MAPI_Tax>
            </fixedCostTax>
            <affiliate>
                <HIPAY_MAPI_Affiliate>
                    <customerId>331</customerId>
                    <accountId>59074</accountId>
                    <val>10.00</val>
                    <percentageTarget>31</percentageTarget>
                </HIPAY_MAPI_Affiliate>
                <HIPAY_MAPI_Affiliate>
                    <customerId>332</customerId>
                    <accountId>59075</accountId>
                    <val>15.00</val>
                    <percentageTarget>26</percentageTarget>
                </HIPAY_MAPI_Affiliate>
            </affiliate>
            <orderTitle>welcome gift</orderTitle>
            <orderInfo>box</orderInfo>
            <orderCategory>5</orderCategory>
        </HIPAY_MAPI_Order>
        <HIPAY_MAPI_Order>
                <shippingAmount>0</shippingAmount>
                <shippingTax></shippingTax>
                <insuranceAmount>5.00</insuranceAmount>
                <insuranceTax>
                        <HIPAY_MAPI_Tax>
                                        <taxName>TVA (5.5)</taxName>
                                <taxVal>5.50</taxVal>
                                <percentage>true</percentage>
                        </HIPAY_MAPI_Tax>
                        <HIPAY_MAPI_Tax>
                                <taxName>TVA (19.6)</taxName>
                                <taxVal>19.60</taxVal>
                                <percentage>true</percentage>
                        </HIPAY_MAPI_Tax>
                </insuranceTax>
                <fixedCostAmount>0.25</fixedCostAmount>
                <fixedCostTax>
                        <HIPAY_MAPI_Tax>
                                <taxName>TVA (5.5)</taxName>
                                <taxVal>5.50</taxVal>
                                <percentage>true</percentage>
                        </HIPAY_MAPI_Tax>
                </fixedCostTax>
                <affiliate>
                        <HIPAY_MAPI_Affiliate>
                                <customerId>331</customerId>
                                <accountId>59074</accountId>
                                <val>10.00</val>
                                <percentageTarget>31</percentageTarget>
                        </HIPAY_MAPI_Affiliate>
                        <HIPAY_MAPI_Affiliate>
                                <customerId>332</customerId>
                                <accountId>59075</accountId>
                                <val>15.00</val>
                                <percentageTarget>26</percentageTarget>
                        </HIPAY_MAPI_Affiliate>
                </affiliate>
                <orderTitle>subscription on review</orderTitle>
                <orderInfo>science</orderInfo>
                <orderCategory>5</orderCategory>
        </HIPAY_MAPI_Order>
</order>
<items>
        <HIPAY_MAPI_Installment>
                <price>5.00</price>
                <tax>
                        <HIPAY_MAPI_Tax>
                                <taxName>TVA (19.6)</taxName>
                                <taxVal>19.60</taxVal>
                                <percentage>true</percentage>
                        </HIPAY_MAPI_Tax>
                </tax>
                <first>true</first>
                <paymentDelay>1H</paymentDelay>
        </HIPAY_MAPI_Installment>
        <HIPAY_MAPI_Installment>
                <price>12.50</price>
                <tax>
                        <HIPAY_MAPI_Tax>
                                <taxName>TVA (19.6)</taxName>
                                <taxVal>19.60</taxVal>
                                <percentage>true</percentage>
                        </HIPAY_MAPI_Tax>
                </tax>
                <first>false</first>
                <paymentDelay>7D</paymentDelay>
        </HIPAY_MAPI_Installment>
        </items>
</HIPAY_MAPI_MultiplePayment>
```

Appendix 4 : XML data feed [B] returned in case of success

Parameter	Description
status	“accepted”
url	URL pointing to the payment page where the customer should be redirected

Example of the data feed returned

```
<?xml version="1.0" encoding="UTF-8"?>
<mapi>
        <mapiversion>1.0</mapiversion>
        <md5content>397cc58864537e8547743349dbd1238c</md5content>
        <result>
                <status>accepted</status>
                <url>https://payment.hipay.com/index/mapi/id/4b840c9421dfc</url>
        </result>
</mapi>
```

Appendix 5 : XML data feed [B] returned in case of error

The information on the error encountered varies according to this chart.

Parameter	Description
status	“ error “
message	Description of the error

Example of the data feed returned:

```
<?xml version="1.0" encoding="UTF-8"?>
<mapi>
        <mapiversion>1.0</mapiversion>
        <md5content>819d143b2d0c11d64b51fbbc0decb7b7</md5content>
        <result>
                <status>error</status>
                <message>MerchantAccount or merchantUserSpace is not exists or disabled - Invalid
        login value : 11111</message>
        </result>
</mapi>
```

Appendix 6 : XML data feed [C] of some notification on the payment

Refer to chapter 19 for further information concerning notifications

Parameter	Description
operation	Type of notification May be authorization / capture / cancellation / refund / reject OR subscription
status	Value of operation ok / nok or cancel

date	Date of the operation execution, in this format: YYYY-MM-DD
time	Time of the operation execution, in this format: HH:MM:SS UTC+0000
transid	Hipay ID for the transaction
SubscriptionId (if the payment is tied to a subscription)	Subscription id
origAmount	The amount of the payment defined by the merchant
origCurrency	The currency of the payment defined by the merchant
IdForMerchant	Merchant ID of the transaction
emailClient	Issuer Email
MerchantDatas	Data posted by the merchant during the intitial XML feed of the payment
refProduct	Products references

All of these data fields are systematically transmitted, except for the SubscriptionId field that will only be sent if the payment is tied to a subscription.

The value of the other fields will depend on the original payment object. Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<mapi>
        <mapiversion>1.0</mapiversion>
        <md5content>c0783cc613bf025087b8bf5edecac824</md5content>
        <result>
                <operation>capture</operation>
                <status>ok</status>
                <date>2010-02-23</date>
                <time>10:32:12 UTC+0000</time>
                <transid>4B83AEA905C49</transid>
                <subscriptionId>753EA685B55651DC40F0C2784D5E1170</subscriptionId> (if the
transaction is attached to a subscription)
                <origAmount>10.20</origAmount>
                <origCurrency>EUR</origCurrency>
                <idForMerchant>REF6522</idForMerchant>
                <emailClient>email_client@hipay.com</emailClient>
                <merchantDatas>
                        <_aKey_id_client>2000</_aKey_id_client>
                        <_aKey_credit>10</_aKey_credit>
                </merchantDatas>
                <refProduct0>REF6522</refProduct0>
        </result>
</mapi>
```

Appendix 7 : Order and product categories

The order or product categories are attached to, and depend upon, the merchant site’s category. Depending on the category that is associated with the site, the categories that are available to the order and products will NOT be the same.
You can obtain the list of order and product category ID’s for the merchant site at this URL:
Live platform : https://payment.hipay.com/order/list-categories/id/[merchant_website_id]
Test platform : https://test-payment.hipay.com/order/list-categories/id/[merchant_website_id]

Returned data (example) :

```
<mapi>
        <categoriesList>
                <category id="200">Label1</category>
                <category id="550">Label2</category>
                <category id="99">Label3</category>
                <category id="205">Label4</category>
        </categoriesList>
</mapi>
```

Appendix 8 : Category of authorized age groups

These are the possible values:
ALL : For all ages.
+12 : For ages 13 and over.
+16 : For ages 16 and over.
+18 : For ages 18 and over.

Appendix 9 : Proxy management and network troubleshooting

If your network is equipped with an entry/exit proxy server, or if you cannot get a response from the Hipay platform, you can try the following options in the [mapi_defs.php] file to determine where the problem is located:

```
// Number of seconds before timing-out with the Hipay server,
// To control the speed of your access to the Hipay platform.
define(‘HIPAY_MAPI_CURL_TIMEOUT’, 30);

// Configuration of the proxy server
// Set this option to TRUE to request the CURL to pass through a proxy
define(‘HIPAY_MAPI_CURL_PROXY_ON’, true);

// Proxy address :
define(‘HIPAY_MAPI_CURL_PROXY’, ‘http://proxy.monreseau.com’);

// Proxy port
define(‘HIPAY_MAPI_CURL_PROXYPORT’,’4040’);

// Configuration of the log file for the CURL, in the event of connection problems with the Hipay server:
define(‘HIPAY_MAPI_CURL_LOG_ON’, true);

// Log file of the CURL (in the Windows environment, the file path should be:
// C:\tmp\mapicurl.log
define(‘HIPAY_MAPI_CURL_LOGFILE’, ‘/tmp/mapicurl.log’);
```

Contact

For any technical problems, please contact us:

technique@hipay.com

Brussels, Belgium
Tel. : (+32) 2 213 07 75
Fax. : (+32) 2 743 47 37

Paris, France
Tel. : (+33) 1.73.03.89.55
Fax : (+33) 1.73.03.89.74

“Buy, Sell and Control With Hipay”