

# Magento Ogone Extension

Ogone GmbH  
Wendelinstrasse 1  
50933 Köln  
HRB 72070 – Amtsgericht Köln  
Geschäftsführer Peter De Caluwe, Koen Casteleyn, Marcus Mosen  
T: +49 (0)221 455 30 180 – F: +49 (0)221 455 30 181  
M: [sales@ogone.com](mailto:sales@ogone.com) – I: [www.ogone.com](http://www.ogone.com)



# Contents

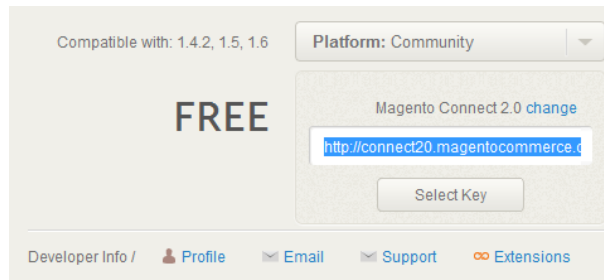
<b>1</b>	<b>Installation</b>	<b>3</b>
1.1	Requirements	3
1.2	Installation process	3
<b>2</b>	<b>Configuration</b>	<b>4</b>
2.1	Ogone Backend	4
2.1.1	Payment methods	4
2.1.2	User management	4
2.1.3	Technical Information	4
2.2	Magento Back Office	8
2.2.1	Payment method configuration	9
2.2.1.1	Open Invoice	9
2.2.1.2	InterSolve	9
2.2.1.3	Recent Payment Information (Ogone Alias Manager)	9
<b>3</b>	<b>Use</b>	<b>11</b>
3.1	General	11
3.1.1	InterSolve	11
3.1.2	Recent Payment Information (Alias Manager)	11
3.2	Shipment receipt	12
3.3	Invoicing	13
3.4	Cancellations	13
3.5	Refunds / Credit Memo	14
3.6	Additional Information	15
3.7	Trigger Payments in Magento Backend (Ogone MOTO eTerminal)	16
3.7.1	Payment using existing Ogone-Transaction	16
<b>4</b>	<b>Notes</b>	<b>17</b>
4.1	General	17
4.2	Multistore	17
4.2.1	Example - different domains using the same Ogone account	17
4.2.2	Example - Store code in url using the the same Ogone account	18
4.2.3	Example - different domains and different Ogone accounts	18
4.2.4	Example - Store code in url and different Ogone accounts	19
<b>5</b>	<b>Workflow</b>	<b>20</b>
5.1	Payment	20

# 1 Installation

## 1.1 Requirements

- PHP 5.2+
- An active Ogone Account
- One of the following Magento versions
  - Community Edition 1.5 - 1.7
  - Enterprise Edition 1.10 - 1.12

## 1.2 Installation process



*Copy the "Extension Key" of the Magento Commerce website*

1. Copy the "Extension Key" of the Magento Commerce website
2. Login to your Magento backend
3. Go to "System" -> "Magento Connect" -> "Magento Connect Manager"
4. Login to your "Magento Connect Manager"
5. Select the tab "Extensions"
6. Copy the key of the Magento Commerce website into the field "Paste extension key to install" and click on "Install"
7. Under some circumstances you need to select "Proceed" to confirm the installation.
8. In the black installation console will the message "Package ... installed successfully" be shown. The module is now correctly installed.

## 2 Configuration

### 2.1 Ogone Backend

Login to your Ogone back office and enter the following settings in the menu.

#### 2.1.1 Payment methods

Depending on the Ogone subscription you have selected it might be necessary to activate further mandatory options in your Ogone account. These options are: OPC, DirectLink (New Payment), DirectLink (Maintenance) and D3D. Please check if these options are activated by default and if this is not the case please contact Ogone for the activation.

The Magento Ogone extension might not work in conjunction with other extensions. To avoid malfunction of the Ogone extension we recommend to disable any other extension for test purposes.

#### 2.1.2 User management

The shop needs a separate API-User to communicate with Ogone. You can create the API-User via "Configuration"-> "Users" select "New User". Select a USERID (for example: APIUSER) and fill in the name and an existing e-mail address. The field "External USERID" can be left empty. Select the profile "Administrator". Select the Access Type "API only". Write down the USERID and password which are shown on your screen.

**Keep in mind that the USERID and the password may not contain any special characters.**

If Ogone creates a password with a special character, please change the password:

- Select Password in the menu on the left
- Select the USERID of the API-User
- Change the password to a password without any special characters

#### 2.1.3 Technical Information

You have to configure the Technical settings in the Ogone Back office via "Configuration" -> "Technical Information" -> "Global Transactionparameters": Select "Direct sale" or "Authorisation" and change the "Payment retries" value if you want to.

Global transaction parameters	Global security parameters	Payment Page layout	Data and origin verification	Transaction feedback	Transaction e-mails	Test Info
<p><b>Default operation code</b></p> <p> <input checked="" type="radio"/> Sale  <input type="radio"/> Authorisation</p> <p>With Ogone e-Terminal, the default operation code you configured above will be valid for all transactions.</p> <p>With Ogone e-Commerce, Ogone DirectLink, Ogone Batch, you can overwrite the default operation code configured above by sending along an operation code in the transaction details.</p> <p><b>Default data capture (payment) procedure</b></p> <p> The default data capture (payment) procedure is only applicable if you have 'Authorisation' as the default operation code for your account or you included the 'Authorisation' operation code in the transaction details.</p> <p><input checked="" type="radio"/> Data capture by the merchant (manual or automatic).  <input type="radio"/> Automatic data capture by our system at the end of the day.  <input type="radio"/> Automatic data capture by our system after <input type="text" value="0"/> days.</p> <p><b>Payment retry</b></p> <p>The number of times you allow a customer to try the payment on the secure Ogone payment pages (with the same or different payment details) within a single order: <input type="text" value="10"/></p> <p><b>Processing for individual transactions</b></p> <p> <input checked="" type="radio"/> Always online (Immediate).  <input type="radio"/> Online but switch to offline in intervals when the online acquiring system is unavailable.  <input type="radio"/> Always offline (Scheduled).</p> <p><b>Default ECI value</b></p> <p> It is vital that the ECI is defined in accordance with your acquirer's specifications. If an incorrect ECI is sent, it could infringe the rules set out in your acquiring agreement.</p> <p><input type="text" value="7 - E-commerce with SSL encryption"/></p> <p>With Ogone e-Commerce, the ECI will be determined dynamically depending on the transaction context.</p> <p>With Ogone e-Terminal, Ogone DirectLink, Ogone Batch, you can overwrite the default ECI value configured above by submitting another ECI value in the transaction details.</p>						

Then go to "Global security parameters" and select the SHA-1 hashing algorithm and UTF-8 character encoding.

Global transaction parameters	Global security parameters	Payment Page layout	Data and origin verification	Transaction feedback	Transaction e-mails	Test Info
<p><b>Hashing method</b></p> <p>This configuration affects all signature checks of which the pass phrases are configured under "Data and origin verification" and "Transaction feedback".</p> <p> Hash algorithm:</p> <p><input checked="" type="radio"/> SHA-1  <input type="radio"/> SHA-256  <input type="radio"/> SHA-512</p> <p> Character encoding:</p> <p><input type="radio"/> Use the character encoding expected with the initial transaction request (depending on the URL called)  <input type="radio"/> ISO-8859-1  <input checked="" type="radio"/> UTF-8</p>						


The settings in the "Payment Page layout" tab are optional.

Global transaction parameters	Global security parameters	Payment Page layout	Data and origin verification	Transaction feedback	Transaction e-mails	Test Info
<p><b>Cancel button</b></p> <p><input type="checkbox"/> Hide the "Cancel" button on the Ogone secure payment pages.</p> <p><b>Back button redirection</b></p> <p> URL of the web page to display to the customer when he clicks the "back" button on our secure payment page.  <input type="text"/></p>						


In the tab "Data and origin verification" do you need to fill in the SHA-IN passphrases. **Please keep in mind that you need to fill the SHA-IN pass phrase for e-Commerce \_and\_ for DirectLink.** SHA-In and SHA-OUT keys have to be alphanumeric only and both keys should have the same value.

Global transaction parameters	Global security parameters	Payment Page layout	<b>Data and origin verification</b>	Transaction feedback	Transaction e-mails	Test Info
-------------------------------	----------------------------	---------------------	-------------------------------------	----------------------	---------------------	-----------

**Checks for e-Commerce**

 URL of the merchant page containing the payment form that will call the page:orderstandard.asp

You may enter several URL, separated by ";".


 SHA-IN Pass phrase

**Checks for Ogone e-Terminal and Ogone Batch (Manual)**


When using these transaction submission modes, you have to log into the back-office beforehand.

**Checks for Ogone DirectLink and Ogone Batch (Automatic)**

When using these transaction submission modes, login details have to be transmitted in the HTTP parameters.

 IP address of the server calling the APIs used for any automatic or direct processing.

You may enter several IP addresses, separated by ";".

 SHA-IN Pass phrase

Please take over the following settings:

Global transaction parameters	Global security parameters	Payment Page layout	<b>Data and origin verification</b>	Transaction feedback	Transaction e-mails	Test Info
-------------------------------	----------------------------	---------------------	-------------------------------------	----------------------	---------------------	-----------

**e-Commerce**

**HTTP redirection in the browser**

URL of the web page to be displayed to the customer after the payment process:


Accepturl: displayed when the payment has been authorized, stored, accepted or is waiting to be accepted.


Declineurl: displayed when the acquirer declines the authorization more than the maximum permissible number of times (as defined in the payment retry section of the Global transaction parameters tab).

Exceptionurl: displayed when the payment result is uncertain.


Cancelurl: displayed when the customer cancels the payment.

You can overwrite the default redirection URLs configured above by sending along redirection URLs in the transaction details.

 ☒ I want to receive transaction feedback parameters on the redirection URLs.

 ☐ I want Ogone to display a short text to the customer on the secure payment page if a redirection to my website is detected immediately after the payment process.

**Direct HTTP server-to-server request**

 Timing of the request

☐ No request.  
☒ Always deferred (not immediately after the payment).  
☐ Always online (immediately after the payment to allow customisation of the response seen by the customer).  
☐ Online but switch to a deferred request in intervals when the online requests fail.

URL of the merchant's post-payment page:

If the payment's status is "accepted", "on hold" or "uncertain".

If the payment's status is "cancelled by the client" or "too many rejections by the acquirer".

Request method

☒ POST  
☐ GET

**Dynamic e-Commerce Parameters**

This exchange box allows you to select or deselect the parameters you wish to receive as feedback for a transaction and move them from one box to another. Please note that NCERROR, PAYID, ORDERID, STATUS are default parameters and selected as such and cannot be removed.

Available	Selected
	AAVCHECK
	ACCEPTANCE
	ALIAS
	CARDNO
	CN
	COMPLUS
	CURRENCY
	CVCCHECK
	ECI
	ED

**General**

☐ I want Ogone to display a "processing" message to the customer during payment processing.

☐ I want Ogone to re-launch the "end of transaction" (post-payment request/redirection) process if required.

**All transaction submission modes**

**Security for request parameters**

☐ SHA-OUT Pass phrase

\*\*\*\*\* Visible

**HTTP request for status changes**

☐ Timing of the request

☐ No request.

☐ Only at the authorisation request of the order.

☒ For each offline status change (payment, cancellation...).

URL on which the merchant wishes to get a deferred HTTP request, should the status of a transaction change offline.

http://www.my-domain.com/ops/api/directlinkpostback

**Please don't forget to configure the SHA-OUT pass phrase.**

Please use the following settings in the "Transaction e-mails" tab.

Global transaction parameters	Global security parameters	Payment Page layout	Data and origin verification	Transaction feedback	Transaction e-mails	Test Info
<p><b>E-mails to the merchant</b></p> <p>E-mail address(es) for transaction-related e-mails:</p> <p>payment@my-domain.com</p> <p>You may enter several e-mail addresses, separated by ";".</p> <p>Receive transaction confirmation e-mails:</p> <p><input type="radio"/> No</p> <p><input type="radio"/> Yes, but only for e-Commerce transactions.</p> <p><input checked="" type="radio"/> Yes, for all transaction submission modes.</p> <p><input type="checkbox"/> Receive e-mails in case of offline transaction status changes:</p> <p><input type="radio"/> No</p> <p><input type="radio"/> Yes, but only at the authorisation request of the order.</p> <p><input checked="" type="radio"/> Yes, for each offline status change (payment, cancellation ...).</p> <p><b>E-mails to the customer</b></p> <p><input type="checkbox"/> I want Ogone to send a transaction confirmation e-mail to the customer.</p> <p><input type="checkbox"/> I want Ogone to send a transaction confirmation e-mail to the customer at the time of the capture.</p> <p><input type="checkbox"/> I want Ogone to send a transaction confirmation e-mail to the customer at the time of the refund.</p>						

Configure the following setting in the "Test info" tab.

Global transaction parameters	Global security parameters	Payment Page layout	Data and origin verification	Transaction feedback	Transaction e-mails	Test Info																		
<b>Test Info</b> <input type="radio"/> I want to simulate transaction results based on the amount. <input checked="" type="radio"/> I want to simulate transaction results based on the card number.																								
<table border="1"> <thead> <tr> <th rowspan="2">TRANSACTION RESULT</th> <th colspan="2">BASED ON AMOUNT</th> <th rowspan="2">BASED ON CARD NUMBER</th> </tr> <tr> <th>Amount</th> <th>Amount x 100</th> </tr> </thead> <tbody> <tr> <td>Successful</td> <td>0 - 9000</td> <td>0 - 900000</td> <td>Any other card number</td> </tr> <tr> <td>Refused</td> <td>&gt; 9000</td> <td>&gt; 900000</td> <td>4111113333333333</td> </tr> <tr> <td>Uncertain</td> <td>9999</td> <td>999900</td> <td>4111116666666666</td> </tr> </tbody> </table>							TRANSACTION RESULT	BASED ON AMOUNT		BASED ON CARD NUMBER	Amount	Amount x 100	Successful	0 - 9000	0 - 900000	Any other card number	Refused	> 9000	> 900000	4111113333333333	Uncertain	9999	999900	4111116666666666
TRANSACTION RESULT	BASED ON AMOUNT		BASED ON CARD NUMBER																					
	Amount	Amount x 100																						
Successful	0 - 9000	0 - 900000	Any other card number																					
Refused	> 9000	> 900000	4111113333333333																					
Uncertain	9999	999900	4111116666666666																					
<b>Test-card numbers</b> <table border="1"> <tbody> <tr> <td>VISA</td> <td>4111 1111 1111 1111</td> </tr> <tr> <td>Visa 3-D Secure</td> <td>4000 0000 0000 0002</td> </tr> <tr> <td>MasterCard</td> <td>5399 9999 9999 9999</td> </tr> <tr> <td>Bancontact/Mister Cash</td> <td>670300000000000003</td> </tr> </tbody> </table>							VISA	4111 1111 1111 1111	Visa 3-D Secure	4000 0000 0000 0002	MasterCard	5399 9999 9999 9999	Bancontact/Mister Cash	670300000000000003										
VISA	4111 1111 1111 1111																							
Visa 3-D Secure	4000 0000 0000 0002																							
MasterCard	5399 9999 9999 9999																							
Bancontact/Mister Cash	670300000000000003																							
<b>Make a test payment</b> <a href="#">Make a test payment with Ogone e-Commerce in ISO</a> <a href="#">Make a test payment with Ogone e-Commerce in UTF-8</a> <a href="#">Make a test payment with Ogone e-Terminal</a> <a href="#">Make a test payment with Ogone DirectLink</a> <a href="#">Make a direct query request</a> <a href="#">Make a direct maintenance request</a> <a href="#">Make a test payment with Ogone Batch</a> <a href="#">Test an automatic file download</a> <a href="#">Test your SHA-1-IN Signature</a>																								
<b>Documentation and test pages</b> <a href="#">Click here for an overview of documentation and test pages.</a>																								

## 2.2 Magento Back Office

After the installation of the module, you now go to the Magento back office -> System -> Configuration -> Payment services. There you will see the option "Ogone Account". Click and fill in the following information:

Setting	Description
PSPID	Fill in the PSPID of your Ogone account.
SHA-IN Pass phrase	Fill in the SHA-IN passphrase that you have setup in your Ogone back office -> Technical information -> Data and Origin verification tab.
SHA-OUT Pass phrase	Fill in the SHA-OUT passphrase that you have setup in the Ogone back office -> Technical information -> Transaction feedback tab.
Gateway URLs	The gateway URLs are already correctly configured. However, when you switch from the TEST environment to the PRODUCTION environment, you need to replace these TEST URLs by PRODUCTION URLs, which are shown in the comment of this configuration fields
API User / API Password	Fill in the API details that you have created in the Ogone Back office.
Payment action	Please keep in mind that this settings needs to be equal to the setting in the Ogone back office -> Technical information -> Global transaction parameters -> Default operation code.
Payment template	When you select <b>Magento</b> , the payment page will be in the same layout as your shop. When you select <b>Ogone</b> , all the fields that follow with reference to the Ogone template are then mandatory to fill in.



Layout of payment methods	Here you can select the way that the payment methods are shown in the checkout page.
Debug	When this option is selected, all requests and answers from Ogone will be checked. We recommend you to activate this option only when you are in test mode and not in production mode.

## 2.2.1 Payment method configuration

After that you should configure your payment methods at System -> Configuration -> Payment Methods.

Please remember, that you should only activate payment methods that are activated in Ogone backend via "Configuration" -> "Payment methods". For activations and further changes please contact Ogone customer service [support@ogone.com](mailto:support@ogone.com).

### 2.2.1.1 Open Invoice

If you enable Open Invoice payments, you must configure your shop to request the customer's birthday and gender. In case of Open Invoice NL also the customers VAT number is required. You can activate these settings at System -> Configuration -> Customer Configuration.

### 2.2.1.2 InterSolve

In case of payment method "InterSolve" it is possible to define several brands, which corresponds to its own Ogones payment method. Please notice, that the brands have to be configured at Ogone before you can use it for your shop. You can assign a title to this brand, which is shown to your customer at the checkout.

*Configuration of InterSolve brands*

### 2.2.1.3 Recent Payment Information (Ogone Alias Manager)

This payment method is a synonym for the Ogone Alias Manager. Your customer can reuse his credit card payment information if this payment method is activated. Payment information can be saved for credit card types with inline payment, only.

*Configuration for Recent Payment Information*

Setting	Description
Enabled	If 'Yes' is selected, your customer can reuse his credit card information used in a recent payment
Show information for guests	If this option is enabled a hint informs your guest customers about the advantages of this payment method, since it's not available to customers who are not logged in.
Title	You can set a custom title, which is shown to your customer.

For further details please refer [Recent Payment Information \(Alias Manager\)](#)

## 3 Use

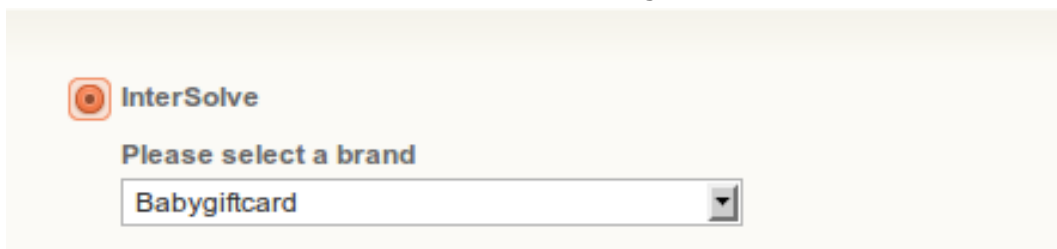
### 3.1 General

This module works like most Magento connections. If you have selected you would like to process credit cards, they will be processed via the Magento platform. However, when 3D Secure is activated, the cardholders will be redirected to an Ogone payment page. This is also applicable for any other payment method where the account holder's details need to be confirmed by the issuer.

When the transaction is processed, you can see the transaction in your Magento back office via Sales -> Orders.

#### 3.1.1 InterSolve

In case of Intersolve payment you can define several brands, which can be selected by the customer during the checkout process. The selected brand will be transmitted to Ogone. There is no selection available, if only one brand is defined. In this case the brand is displayed to the customer and will be transmitted to Ogone.

The screenshot shows a light beige rectangular box. At the top left is an orange square icon with a white circle inside. To its right is the text "InterSolve" in a bold, dark blue font. Below this is the text "Please select a brand" in a smaller, dark blue font. Underneath is a white rectangular dropdown menu with a thin grey border. The text "Babygiftcard" is visible inside the dropdown, and a small downward-pointing arrow is on the right side of the menu.

*Selection of InterSolve brands*

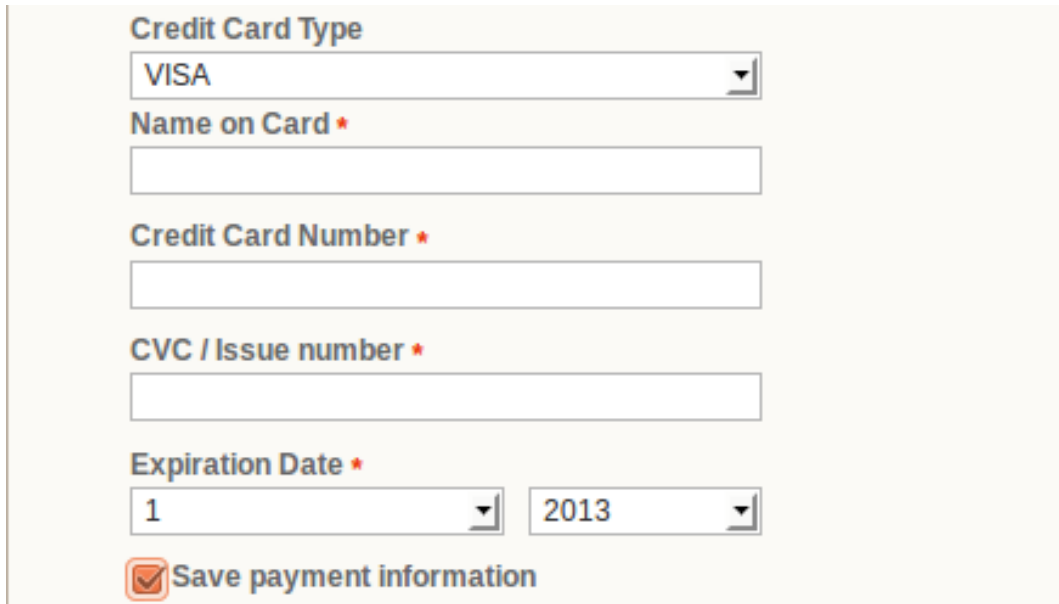
#### 3.1.2 Recent Payment Information (Alias Manager)

If your customer is logged in and the payment method "Recent Payment Information" is activated, the customers payment information are provided to the customer during checkout.

The screenshot shows a light beige rectangular box. At the top left is an orange square icon with a white circle inside. To its right is the text "Recent payment information" in a bold, dark blue font. Below this is a white rectangular box with a thin grey border. Inside this box, the text "Credit Card XXXXXXXXXXXX1111 (VISA)" is displayed in a dark blue font, and below it, the text "expires: 0117" is displayed in a smaller, dark blue font.

*Example of displayed recent payment to your customer*

The selection of the displayed payment information depends on billing address and shipping address in order to prevent abuse. If your customer wants to save his payment information, he has to select the checkbox 'save payment information' on credit card payment section.



The image shows a checkout form for credit card payment. It includes the following fields and elements:

- Credit Card Type:** A dropdown menu with 'VISA' selected.
- Name on Card \***: A text input field.
- Credit Card Number \***: A text input field.
- CVC / Issue number \***: A text input field.
- Expiration Date \***: Two dropdown menus for month and year. The month is set to '1' and the year is set to '2013'.
- Save payment information:** A checkbox that is checked, with the text 'Save payment information' next to it.

*Creation of recent payment information in the checkout*

In addition your customer can list and delete his payment information in the section 'payment information' within his account.

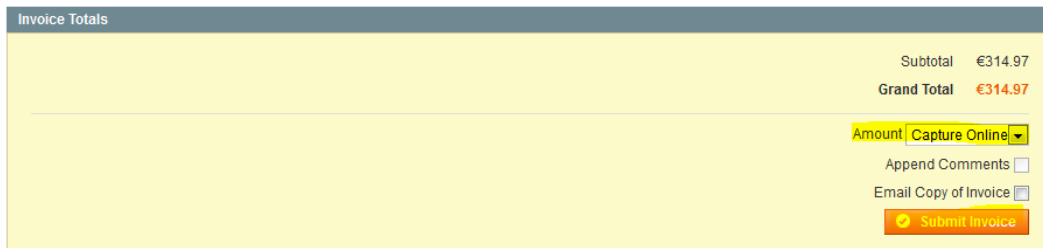
This opportunity is also available in the customers administration section in the Magento backend. Select the customer and then select 'payment information' for a list of stored payment information.

## 3.2 Shipment receipt

The creation of a shipment receipt is made without a connection to the payment service and therefore behaves like Magento-Standard.

### 3.3 Invoicing

Whether you have choose "Authorization" as "Payment Action" in the Magento configuration, an invoice has to be created for payment methods like credit card to trigger the capture process by Ogone. For this purpose click in the order view on button "Invoice" and choose "Capture Online" as amount and finish the invoice creation with "Submit Invoice".



Depending of the payment method the invoice will be created instantly or after a delay by a gateway-response triggered by Ogone.

### 3.4 Cancellations


To proceed a cancel (in case the authorisation was successful) use "Void" instead of "Cancel" in the order view. "Cancel" could not be used because of gateway workflow reasons. Depending of the payment method the cancel would be proceeded instantly or after a delay by a gateway-response triggered by Ogone.

If the cancellation was successful processed by Ogone it can be necessary in some Magento-versions to click on "Cancel" in the order view to set the correct cancelled status in Magento.


The order can be cancelled directly by the button "Cancel", if the order has the state "Pending Payment" and Ogone-status 0 or empty. In this case no gateway request to Ogone will be sent and the product amount will be put back in stock. *Warning:* Please check the payment status in the back-office of Ogone before. By cancelling this order you won't be able to update the status in Magento anymore.

### 3.5 Refunds / Credit Memo

To create a refund firstly you have to open the order view and open the "invoices"-tab in the left menu panel. Choose the invoice you want to create a refund for. In the "invoice view" click on button "Credit Memo". In this form you can adjust the amount to refund and you can also decide if you want to close the Ogone transaction. In order to achieve this, click on the checkbox 'Close Ogone transaction'. Finally confirm the refund by clicking "Refund" button and not "Refund Offline"! The figure below shows the corresponding input fields:

Refund Totals	
Subtotal (Excl.Tax)	€8.00
Subtotal (Incl.Tax)	€8.00
Refund Shipping (Excl. Tax)	<input type="text" value="1"/>
Adjustment Refund	<input type="text" value="0"/>
Adjustment Fee	<input type="text" value="0"/>
<b>Grand Total (Excl.Tax)</b>	<b>€9.00</b>
Total Tax	€0.00
<b>Grand Total (Incl.Tax)</b>	<b>€9.00</b>
Last refund (Close payment transaction) <input type="checkbox"/>	
<div> If this checkbox is marked, the refund will be final: the transaction will be closed and it will not be possible to perform any further refund for this transaction.</div>	
Append Comments <input type="checkbox"/>	
Email Copy of Credit Memo <input type="checkbox"/>	
<div><input checked="" type="button" value="Refund Offline"/> <input checked="" type="button" value="Refund"/></div>	

You can repeat this procedure until the Ogone transaction is closed, that means either the complete amount is already refunded or you have decided to close the Ogone transaction manually by the checkbox 'Close Ogone transaction' before. In this case only offline refunds are available and by creation of the Credit Memo the following message appears:

 **Only offline refund is available, since Ogone transaction is already closed.**

#### **New Credit Memo for Invoice #100000171**

Depending of the payment method the refund would be proceeded instantly or after a delay by a gateway-response triggered by Ogone.

The button "Credit Memo" in the "order view" just creates an internal refund and triggers no refund action by Ogone.

For some payment methods (e.g. iDEAL), you will have to enable refunds in your Ogone account.

## 3.6 Additional Information

In the "Payment Information" box on the "Order View" Page (-> Sales -> Orders -> Click "View" -> Select "Information"), you can find additional payment related information about the order. The information includes payment method type, e.g. VISA credit card, Payment ID, recent payment status and currency.

Secondly, you will find in the lower part of the "Order View" Page in the "Comments History" box a chronologically sorted course of payment status changes. This additional information helps to track payment process changes, e.g. to allow you to verify that a credit card authorisation has changed to a payment capture.

## 3.7 Trigger Payments in Magento Backend (Ogone MOTO eTerminal)

According to Ogone MOTO eTerminal some payment methods are available for order creation in Magento backend, too. Since there are no redirect payment methods allowed, you can't use such methods. Even usage of 3D secure ("Verified by VISA", "MasterCard Secure Code") is not possible.

Enable for backend	Yes	[GLOBAL]
--------------------	-----	----------

*Configuration for MOTO eTerminal features for credit cards and direct debits*

Hint: Magento sends some data to the shop immediately after input. So it may appear, that you have to click on the order submit button twice.

### 3.7.1 Payment using existing Ogone-Transaction

Using this payment method, which is only available in the Magento backend, you can enter the pay id of an existing Ogone transaction directly. In this case no request is triggered to Ogone and all state changes have to be done manually.



## 4 Notes

### 4.1 General

#### **No credits in Ogone Back office**

Only send your credit requests via the Magento back office.

#### **3D-Secure (MasterCard, SecureCode and Verified by Visa) must be activated**

Your account must accept 3D-Secure.

### 4.2 Multistore

The module is suitable to be used in Magento-Multistores with

- multiple websites
- multiple stores
- multiple domains
- multiple Ogone accounts

In order to use multiple Ogone-accounts, you have to follow the steps mentioned in sections [Ogone Backend](#) and [Magento Back Office](#) for each Ogone-account and website or store.

The most important part of the Multishop configuration concerns the correct setup of the response-urls in Ogone. You can find this configuration-fields in the Ogone-backend at "Configuration" -> "Technical information" -> "Transaction feedback"

- Direct HTTP server-to-server request
- HTTP request for status changes.

The correct configuration of this urls should be clarified with the help of the following four examples.

#### 4.2.1 Example - different domains using the same Ogone account

The Multistore consists of two stores which are assigned to different websites.

- Store A operates under domain  
<http://www.my-shop-a.com/>
- Store B operates under domain  
<http://www.my-shop-b.com/>

The correct adjustment in the Ogone-backend would be the following:

- Direct HTTP server-to-server request:  
<http://www.my-shop-a.com/ops/api/postBack>
- HTTP request for status changes:  
<http://www.my-shop-a.com/ops/api/directLinkPostBack>

The feedback will be processed correctly for store B also if the response is send only to the url of store A.

#### 4.2.2 Example - Store code in url using the the same Ogone account

The Multistore consists of two stores which are assigned to the same website. The example would be also valid in case that both stores were assigned to different websites.

- Store A operates under domain  
<http://www.my-shop.com/shopa/>
- Store B operates under domain  
<http://www.my-shop.com/shopb/>

The correct adjustment in the Ogone-backend would be the following:

- Direct HTTP server-to-server request:  
<http://www.my-shop.com/shopa/ops/api/postBack>
- HTTP request for status changes:  
<http://www.my-shop.com/shopa/ops/api/directLinkPostBack>

As before the feedback will be processed correctly for store B also if the response is send only to the url of store A.

#### 4.2.3 Example - different domains and different Ogone accounts

The multistore consists of two stores which are assigned to different websites and use two different Ogone accounts.

- Store A operates under domain  
<http://www.my-shop-a.com/>
- Store B operates under domain  
<http://www.my-shop-b.com/>

The correct adjustment for Store A in the Ogone-backend would be the following:

- Direct HTTP server-to-server request:  
<http://www.my-shop-a.com/ops/api/postBack>
- HTTP request for status changes:  
<http://www.my-shop-a.com/ops/api/directLinkPostBack>

The correct adjustment for Store B in the Ogone-backend would be the following:

- Direct HTTP server-to-server request:

<http://www.my-shop-b.com/ops/api/postBack>

- HTTP request for status changes:

<http://www.my-shop-b.com/ops/api/directLinkPostBack>

#### 4.2.4 Example - Store code in url and different Ogone accounts

The Multistore consists of two stores which are assigned to the same website and use different Ogone accounts. The example would be also valid in case that both stores were assigned to different websites.

- Store A operates under domain

<http://www.my-shop.com/shopa/>

- Store B operates under domain

<http://www.my-shop.com/shopb/>

The correct adjustment for Store A in the Ogone-backend would be the following:

- Direct HTTP server-to-server request:

<http://www.my-shop.com/shopa/ops/api/postBack>

- HTTP request for status changes:

<http://www.my-shop.com/shopa/ops/api/directLinkPostBack>

The correct adjustment for Store B in the Ogone-backend would be the following:

- Direct HTTP server-to-server request:

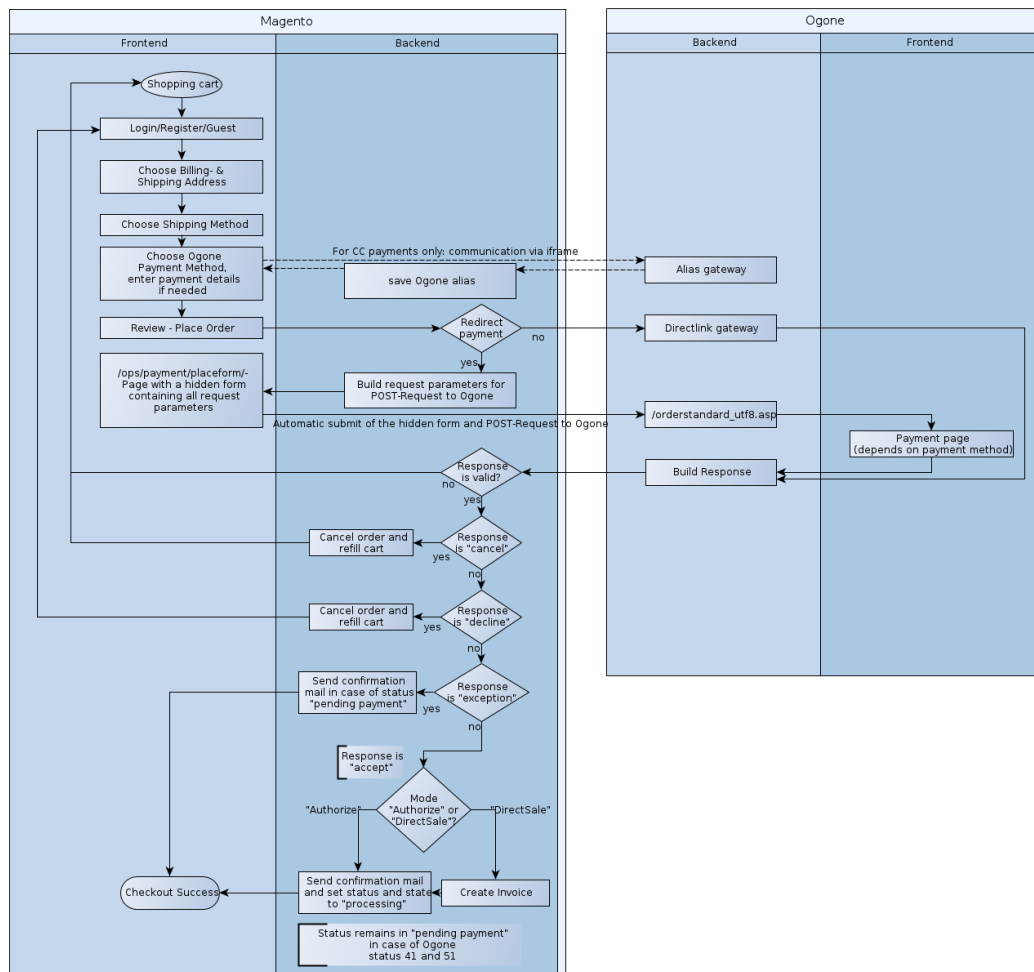
<http://www.my-shop.com/shopb/ops/api/postBack>

- HTTP request for status changes:

<http://www.my-shop.com/shopb/ops/api/directLinkPostBack>

## 5 Workflow

### 5.1 Payment



*Payment Workflow of the Ogone Module*