



Protocol Specifications

« Call & Pay[®] »

IVR Payment Gateway

Version 1.0.2 – Pay per Call

30. June 2009

Copyright © 2009 HORISEN Communications GmbH



Protocol Specifications for the IVR Payment Gateway «Call & Pay®»

1. Revision history

Version	Date	Description	Author
1.0.2	30. June 2009	Documentation and examples update	HORISEN Communications GmbH
1.0.1	06. May 2009	Ajax method implemented	HORISEN Communications GmbH
1.0.0	06. Feb 2009	On/Off and Credit added	HORISEN Communications GmbH
0.9.4	15. Oct 2008	Connection price added	HORISEN Communications GmbH

2. Definitions

Expression	Description
Customer	Party that sells online services, products or downloadable content
Buyer	Party that buys services, products or content
Payment Gateway Provider	Provider of website for payment over the IVR payment system «Call&Pay»
customer.web.site	in examples below, URL of customer website
Transaction ID	Transaction ID displayed to the buyer; to be used in contacts with support
Reference ID	ID that is used in forming URL's to the «Call&Pay» server
Service Kind	Kind of service (online payment, ON/OFF by phone, credit tanking)
Announced code	Code announced to the buyer by «Call&Pay» service during the phone call
CAPTCHA code	Code consisting of randomly selected group of letters and numbers which are displayed as a distorted image.

3. Types of service

Following section describes the three types of service (in following text «service kinds») currently available with «Call&Pay»:

1. Pay per call
2. ON/OFF by phone call (referenced in rest of this document as «on/off service»)
3. Credit tanking (referenced in rest of the text as «credit service»)

3.1 Pay per call

This section describes the pay per call process.

3.1.1 Goal

This service may be used for customers that want to sell online any kind of products, services or e.g. downloadable content, such as video clips, audio files, etc. Customer website provides the purchase price for the product and the «Call&Pay» service selects automatically a drop charge or a per minute based phone number that user has to call to pay the product.

3.1.2 Basic Call Flow

1. Buyer selects a product to buy online and clicks to checkout.
2. Customer website starts transaction by invoking HTTP request to Payment Gateway.
3. Customer website opens popup window with URL described below. Popup displays phone number to call.
4. Buyer makes phone call and code announced has to be entered in the popup, along with CAPTCHA code.
5. Web browser of buyer is redirected to content download page in new popup window.
6. Customer website on `content_link` checks validity of `content_link` redirection.
7. Buyer is allowed to download content.



4. Pictures

Figure 1: Pay per call - sequence diagram in case of usage of a drop charge number

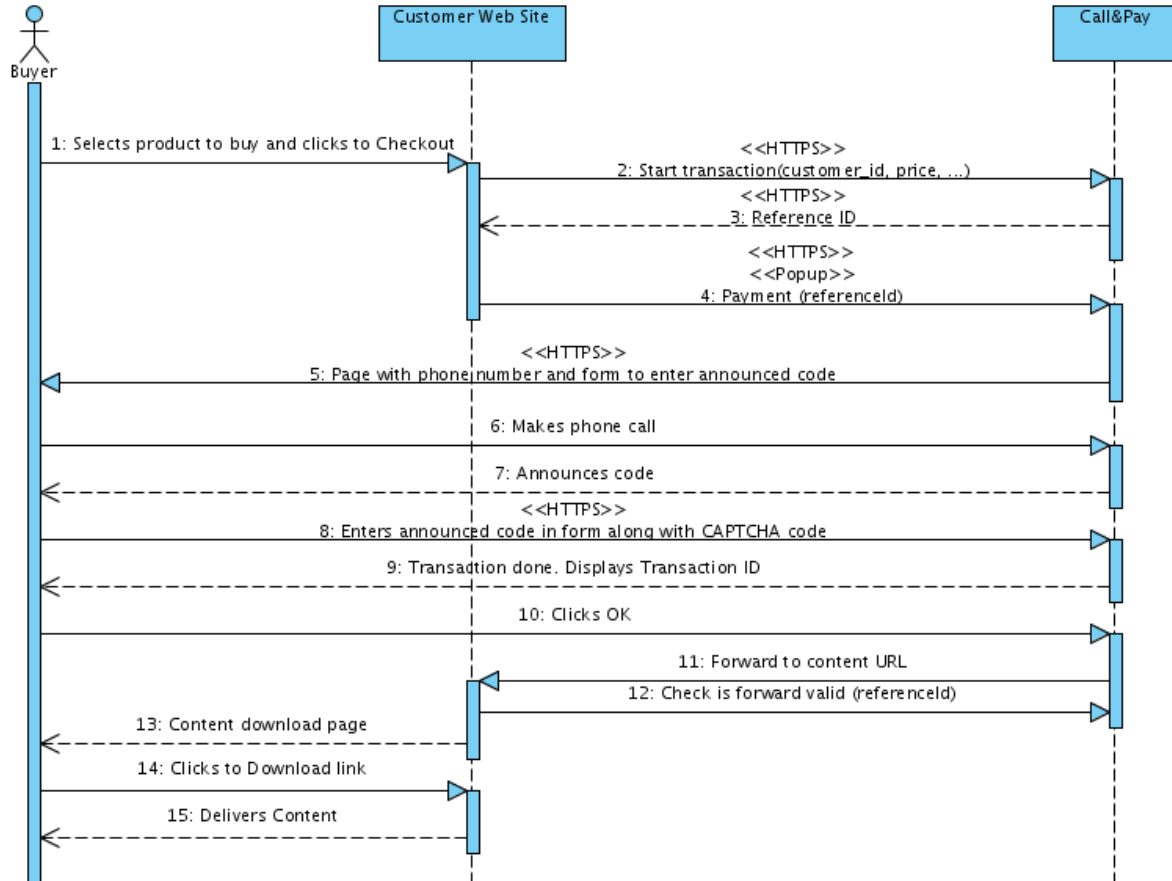
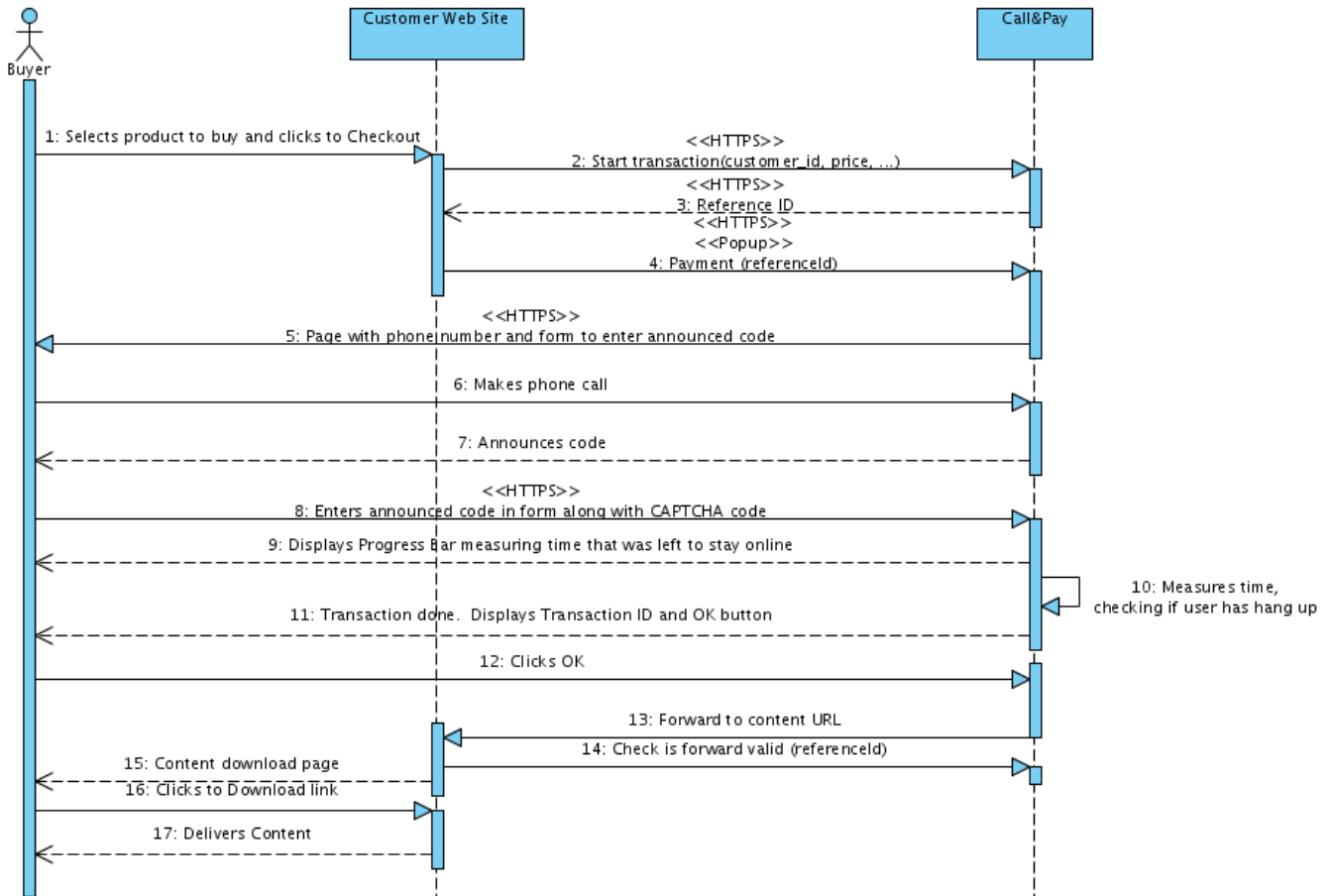


Figure 2: Pay per call - sequence diagram in case of usage of a perminute number



5. Implementation steps

First decision you have to make is how you want content link to be displayed. Due to security limitations of the different browsers (IE, Firefox, Safari, Opera, Chrome), two separate mechanisms are provided:

1. Content link will be displayed in popup window – in further text referred as ‘content popup’
2. Content link will be displayed in host window of web site (where BUY button is) – in further text referred as ‘ajax’.

Depending of decision, following two sections (§5.1 and §5.2) describe these steps in detail.

5.1 Implementation steps for ‘content popup’

1. Set connection parameters in CallAndPay.php
2. Opening popup window for online transaction
3. Start transaction
4. Implement content link (for ‘content popup’).

5.1.1 Set connection parameters in CallAndPay.inc.php

Copy to your web site CallAndPay.inc.php (from example archive).

Edit the file CallAndPay.inc.php and set parameters in class CallAndPay_Config:



Variable	Value
THIS_SITE	Points to URL prefix for example application. For example, if URL prefix is <code>customer.web.site/subdir</code> set this to <code>'http://customer.web.site/subdir'</code>
THIS_SITE_ID	Fully Qualified Domain Name of web site where example site is installed, e.g.: <code>'customer.web.site'</code>
CUSTOMER_ID	Customer ID you got from «Call&Pay» when opening account
PASSWORD	Password you got from «Call&Pay» when opening account
SUPPORT_EMAIL	Set your support e-mail address here
SUCCESS_EMAIL	Set e-mail address to which e-mail will be sent whenever transaction is completed. If set to empty string, no email will be sent.
POPUP_NAME	Name of javascript popup window for content download link. Only alphanumeric characters are allowed. Example: <code>'contentDownload'</code>
POPUP_PARAMS	Javascript parameters of popup window for content download link. Example: <code>'height=300,width=200'</code> . Every parameter of <code>window.open</code> Javascript function is allowed.

5.1.2 Opening popup window for online transaction

«Call&Pay» needs to be started in a popup window.

Copy to your web site `callandpay.php` (from example archive).

Open «checkout» or «buy» page like this (example uses smarty template engine):

```
<?php
require_once('libs/Smarty.class.php');
require_once('CallAndPay.inc.php');
...
$callandpay_config = new CallAndPay_Config();
$smarty->assign('THIS_SITE', $callandpay_config->THIS_SITE);
$smarty->display('product.tpl');
?>
```

In your HTML page that contains the «BUY» button (`ajax-product.tpl`) put in header this code...

```
<head>
...
<script language="JavaScript" src="callandpay.php"></script>
...
</head>
```

...and on click to BUY button open popup using this function:

```
openCallAndPayPopup(url, popup_name);
```

For url parameter put URL to page for starting transaction (see §5.1.3).



5.1.3 Starting transaction

Create one PHP page that does the following:

```
// Create CallAndPay object and create session ID. Then remember this object in session variable
callandpay_config = new CallAndPay_Config();
$callandpay_config->POPOPUP_NAME = 'nocontent';
$callandpay = new CallAndPay($callandpay_config);
$_SESSION['CALLANDPAY'] = $callandpay;
$session_id = $callandpay->createSessionId();

// Configure content link (link that displays page where buyer will be able to download content)
$content_link = $callandpay_config->THIS_SITE . "/download.php";

// Put here list of additional parameters that your web site on content_link
// can use to identify which content is to be downloaded
$parameters = "param1=" . urlencode('value1')
    . "&param2=" . urlencode('value2')
    . "&param3=" . urlencode('value3')
    . "&productName=" . urlencode('Demo MP3 File')
    . "&productId=" . urlencode('12345678')
    . "&paymentType=" . urlencode('phoneCall');

// Start PayPerCall transaction
if ($callandpay->startPayPerCall($price, '000', $currency, $country, $language, $service_class,
    $session_id, $content_link, $parameters))
{
    $callandpay->payment();
}
else
{
    // Display an error in popup
    // e.g.
    // $smarty->assign("error message", "ERROR: " . $callandpay->getLastErrorCode() . " " . $callandpay->getLastErrorMessage());
    // $smarty->display('error.tpl');
}
}
```

This example uses call `$callandpay » startPayPerCall`:

```
public function startPayPerCall($price, $price_cn, $currency, $country, $language, $service_class, $session_id, $content_link, $parameters, $event_url = null, $event_params = null)
```

Parameters are described in next section.

For example implementation see §5.1.5, file checkout.php.



5.1.3.1 Parameters for transaction creation

These parameters need to be set in every «start transaction» invocation.

Variable	Value / Description
price	Setup price has to be given in cents (as PHP string with leading zeros). Example: 1.50 EUR → '150', 0.50 EUR → '050'
price_cn	Set it to '000'
customer_id	Customer ID (provided by Payment Gateway Provider)
country	in 2 letter format. Example: CH, DE, IT, AT, UK, etc.
currency	in 3 letter format. Example: CHF, EUR, etc.
language	Popup window language in 2 letter format. Example: EN, DE, IT, ES, etc.
content_link	Put here URL that displays page where buyer will be able to download content
parameters	Additional parameters for content_link (see example in §5.1.3)
service_class	Possible values: «normal», «adult» and «test». See §5.1.3.2

5.1.3.2 Services classes

There are 3 service classes described in the table below:

Service Class	Description
test	This service class should be used while implementing «Call&Pay» services. «Call&Pay» will always show Swiss fixnet number instead of a VAS number. These calls are visible in statistics with price 0 and kickback 0 when used as drop charge numbers or with connection price 0.2 and kickback 0 when used as per minute numbers, also for «onoff» and «credit» services. This allows you to test the service implementation without having high testing costs. See §6.2
normal	This should be used in production, for non-erotic services.
adult	This should be used in production, for erotic services.

5.1.3.3 Error CODES

Please, check §6.1

5.1.4 Implement content link

Implement one php page that will be opened after payment transaction is successfully completed which will contain link to content that is sold in this transaction. Set 'content_link' and parameters' parameters in §5.1.3 to point to this PHP page and to provide enough information to discover which content is sold in transaction (e.g. song name or file id or similar). In 'parameters' you can set whatever additional parameter your site needs for content link.



```

<?php
require_once("config.inc.php");

$reference_id = $_GET['id'];
// ...
// Get other parameters set in 'parameters' in §5.1.3
//...
$callandpay_config = new CallAndPay_Config();
$callandpay = new CallAndPay($callandpay_config, $reference_id);

// Check validity of content link (optional but recommended)
$check_url = $callandpay->config->THIS_HOSTNAME . $_SERVER['REQUEST_URI'];
if ($callandpay->check($check_url)) {
    // Success. Display download page
    error_log("Transaction completed\n");
    $smarty->display('download.tpl');
} else {
    // Content link validity failed. Display error
    $smarty->assign("error message", "ERROR: " . $callandpay->getLastErrorCode() . " " . $callandpay->getLastErrorMessage());
    $smarty->display('error.tpl');
}

```

For example implementation see §5.1.5, file download.php.

5.1.5 Complete example

Archive is distributed with this document.

This archive contains example customer web site written in PHP that you can install and try.

Example uses third party libraries:

1. smarty template engine (<http://www.smarty.net>)

Files included in example:

- callandpay.php (this file should be copied on production web site also)
- CallAndPay.inc.php (this file should be copied on production web site also, and set as described in §5.1.1)
- index.php is example for your 'checkout' or 'buy' page
- templates/product.tpl is smarty template displayed by index.php
- checkout.php is url parameter as described in §5.1.2 when calling openCallAndPayPopup)
- download.php is download URL as described in §5.1.4
- config.inc.php - configuration settings

5.1.5.1 Installation

Please unpack archive and install it on your test location on the web.

Open CallAndPay.php file in editor and change parameters to match your configuration as described in comments and in §5.1.1.



5.2 Implementation steps for 'ajax'

1. Set connection parameters in CallAndPay.php
2. Opening popup window for online transaction
3. Start transaction
4. Implement signaling mechanism (for 'ajax').

5.2.1 Set connection parameters in CallAndPay.inc.php

Copy to your web site CallAndPay.inc.php (from example archive).

Edit the file CallAndPay.inc.php and set parameters in class CallAndPay_Config:

Variable	Value
THIS_SITE	Points to URL prefix for example application. For example, if URL prefix is customer.web.site/subdir set this to 'http://customer.web.site/subdir'
THIS_SITE_ID	Fully Qualified Domain Name of web site where example site is installed, e.g.: 'customer.web.site'
CUSTOMER_ID	Customer ID you got from «Call&Pay» when opening account
PASSWORD	Password you got from «Call&Pay» when opening account
SUPPORT_EMAIL	Set your support e-mail address here
SUCCESS_EMAIL	Set e-mail address to which e-mail will be sent whenever transaction is completed. If set to empty string, no email will be sent.

5.2.2 Opening popup window for online transaction

«Call&Pay» needs to be started in a popup window.

Copy to your web site callandpay.php (from example archive).

Open «checkout» or «buy» page like this (example uses smarty template engine):

```
<?php
require_once('libs/Smarty.class.php');
require_once('CallAndPay.inc.php');
...
$callandpay_config = new CallAndPay_Config();
$smarty->assign('THIS_SITE', $callandpay_config->THIS_SITE);
$smarty->display('ajax-product.tpl');
?>
```

In your HTML page that contains the «BUY» button (ajax-product.tpl) put in header this code...

```
<head>
...
<script language="JavaScript" src="callandpay.php"></script>
...
</head>
```

...and on click to BUY button open popup using this function:

```
openCallAndPayPopup(url, popup_name);
```

For url parameter put URL to page for starting transaction (see §5.3).



5.2.3 Starting transaction

Create one PHP page that does the following:

```
// Create CallAndPay object and create session ID. Then remember this object in session variable
callandpay_config = new CallAndPay_Config();
$callandpay_config->POPOPUP_NAME = 'nocontent';
$callandpay = new CallAndPay($callandpay_config);
$_SESSION['CALLANDPAY'] = $callandpay;
$session_id = $callandpay->createSessionId();

// Set the link for event signaling
$event_url = $callandpay_config->THIS_SITE . "/ajax-signal.php";
$event_params = 'evparam1=' . urlencode('evvalue1') . '&evparam2=' . urlencode('evvalue2');

// Start PayPerCall transaction
if ($callandpay->startPayPerCall($price, '000', $currency, $country, $language, $service_class,
    $session_id, "", "", $event_url, $event_params))
{
    // TODO: insert in database info that transaction
    // with Reference ID $callandpay->getReferenceId() is started
    // e.g.
    // $sql = "INSERT INTO paypercall_status (transaction_id, status, insert_dt, status_dt)"
    //      . " VALUES (?, 'C', now(), now())";
    // $dbconn->Execute($sql, array($callandpay->getReferenceId()));

    $callandpay->paymentAjax();
}
else
{
    // Display an error in popup
    // e.g.
    // $smarty->assign("error_message", "ERROR: " . $callandpay->getLastErrorCode() . " " . $callandpay->getLastErrorMessage());
    // $smarty->display('error.tpl');
}
}
```

This example uses call `$callandpay » startPayPerCall`:

```
public function startPayPerCall($price, $price_cn, $currency, $country, $language, $service_class, $session_id, $content_link, $parameters, $event_url = null, $event_params = null)
```

Parameters are described in next section.

For example implementation see §5.2.5, file `ajax-checkout.php`.



5.2.3.1 Mandatory parameters for transaction creation

These parameters need to be set in every «start transaction» invocation.

Variable	Value / Description
price	Setup price has to be given in cents (as PHP string with leading zeros). Example: 1.50 EUR → '150', 0.50 EUR → '050'
price_cn	Set it to '000'
currency	in 3 letter format. Example: CHF, EUR, etc.
content_link	(for AJAX variant set it to empty string)
parameters	(for AJAX variant set it to empty string)
support_email	Customer provided Email address for contact in case of problems
language	Popup window language in 2 letter format. Example: EN, DE, IT, ES, etc.
service_class	Possible values: «normal», «adult» and «test». If not provided, default is «normal». See §5.2.3.2
event_url	Signal URL (see §5.2.4)
event_parameters	Optional. URL encoded parameters sent to event_url (see §5.2.4).

5.2.3.2 Services classes

There are 3 service classes described in the table below:

Service Class	Description
test	This service class should be used while implementing «Call&Pay» services. «Call&Pay» will always show Swiss fixnet number instead of a VAS number. These calls are visible in statistics with price 0 and kickback 0 when used as drop charge numbers or with connection price 0.2 and kickback 0 when used as per minute numbers, also for «onoff» and «credit» services. This allows you to test the service implementation without having high testing costs. See §6.2
normal	This should be used in production, for non-erotic services.
adult	This should be used in production, for erotic services.

5.2.3.3 Error CODES

Please, check §6.1.

5.2.4 Implement signalling mechanism

Implement one PHP file that will accept signals sent by «Call&Pay» system (signal mentioned in §5.2.3.1)

URL is called with parameters provided by signal_parameters. Additional parameters provided by Call And Pay application:

Variable	Value / Description
Id	Reference ID for transaction provided as a response to 'start transaction' method, see §5.2.3.
event	'completed' or 'interrupted'

For example implementation see §5.2.5, file ajax-signal.php and ajax-checkPaymentStatus.php.

5.2.4.1 Completed event

This event is sent when user completes transaction (he payed for content).

5.2.4.2 Interrupted event

Event 'interrupted' is sent when user makes a call, but in the case of perminute number he does not wait requested time and hangs up before full price is charged.



5.2.5 Complete example

Archive is distributed with this document.

This archive contains example customer web site written in PHP that you can install and try.

Example uses third party libraries:

2. smarty template engine (<http://www.smarty.net>)
3. adodb (<http://adodb.sourceforge.net>)
4. prototype (<http://www.prototypejs.org>)

Files included in example:

- `callandpay.php` (this file should be copied on production web site also)
- `CallAndPay.inc.php` (this file should be copied on production web site also, and set as described in §5.2.1)
- `index.php` is example for your 'checkout' or 'buy' page
- `templates/ajax-product.tpl` is smarty template displayed by `index.php`
- `ajax-checkout.php` is url parameter as described in §5.2 when calling `openCallAndPayPopup`
- `ajax-signal.php` is signaling URL as described in §5.4
- `ajax-checkPaymentStatus.php` is AJAX is method used by your 'checkout' or 'buy' page to check status of transaction (see also `templates/ajax-product.tpl`)
- `config.inc.php` - configuration settings
- `connectionfolder/ado_config.inc.php` - database connection parameters

5.2.5.1 Installation

Please unpack archive and install it on your test location on the web.

Open `CallAndPay.php` file in editor and change parameters to match your configuration as described in comments and in §5.2.1.

5.2.5.2 Create example database

Create one MySQL database (or use existing) and use script `db/create.sql` to create table needed by example. Create database user with password, and set connection parameters in `connectionfolder/ado_config.inc.php`



6. Implementation notices

6.1 Error Codes

HTTP invocation of «start» and «check» methods may return error. This is list of error codes:

Error Code	Message
90	HTTP error
91	Unknown error (connection problems)
101	Price not supported
102	Unknown country %s
103	Creating transaction failed due to system error on «Call&Pay» service
104	No transaction with ID=%s
105	Illegal content link URL
106	Unknown currency %s
107	Unknown customer %s
108	Wrong password for customer %s

6.2 Development and test phase

During development and initial test phase, please use:

Variable	Value
service_class	«test»
country	«CH»
currency	«CHF»
price	«000»
price_cn	«020» for «onoff» and «credit» service or 0 for «pay per call» service

System will then use phone numbers that can be called for testing, for price of standard fixnet call to Switzerland.

6.3 Switching to production configuration

Replace service_class with from test to normal for non-erotic services or adult for erotic services. Also, please use in production real prices.

