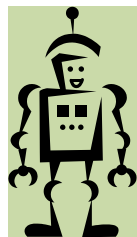
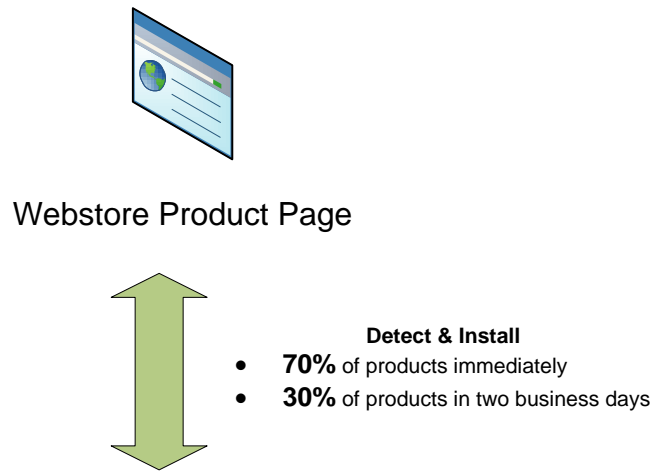


# Automatic Installation of Maybe3D Products

Technical Specification

2<sup>nd</sup> February, 2011  
Maybe3D LLC

# Overview



Syntax  
Robot



Human Assistance  
&  
Quality Assurance



Maybe3D Employee

## Automatic Installation

The technique of Maybe3D Automatic Installation System is based on frequency analysis of the occurrences of product names in the product page.

Most part of the work (about 70%) is done by a special robotic web service. It downloads the product web page, and determines which product Maybe3D must be installed on it. The answer is given almost immediately. In subsequent requests robot gives the cached results.

If the Robot has doubts about the correctness of detection, it leaves a message for an Maybe3D employee about the necessity of verification and correction.

Within two business days all products will be installed.

All the product pages are checked by the Maybe3D Quality Assurance Dep.

## Methods of Automatic Installation

### Using Starter Utility

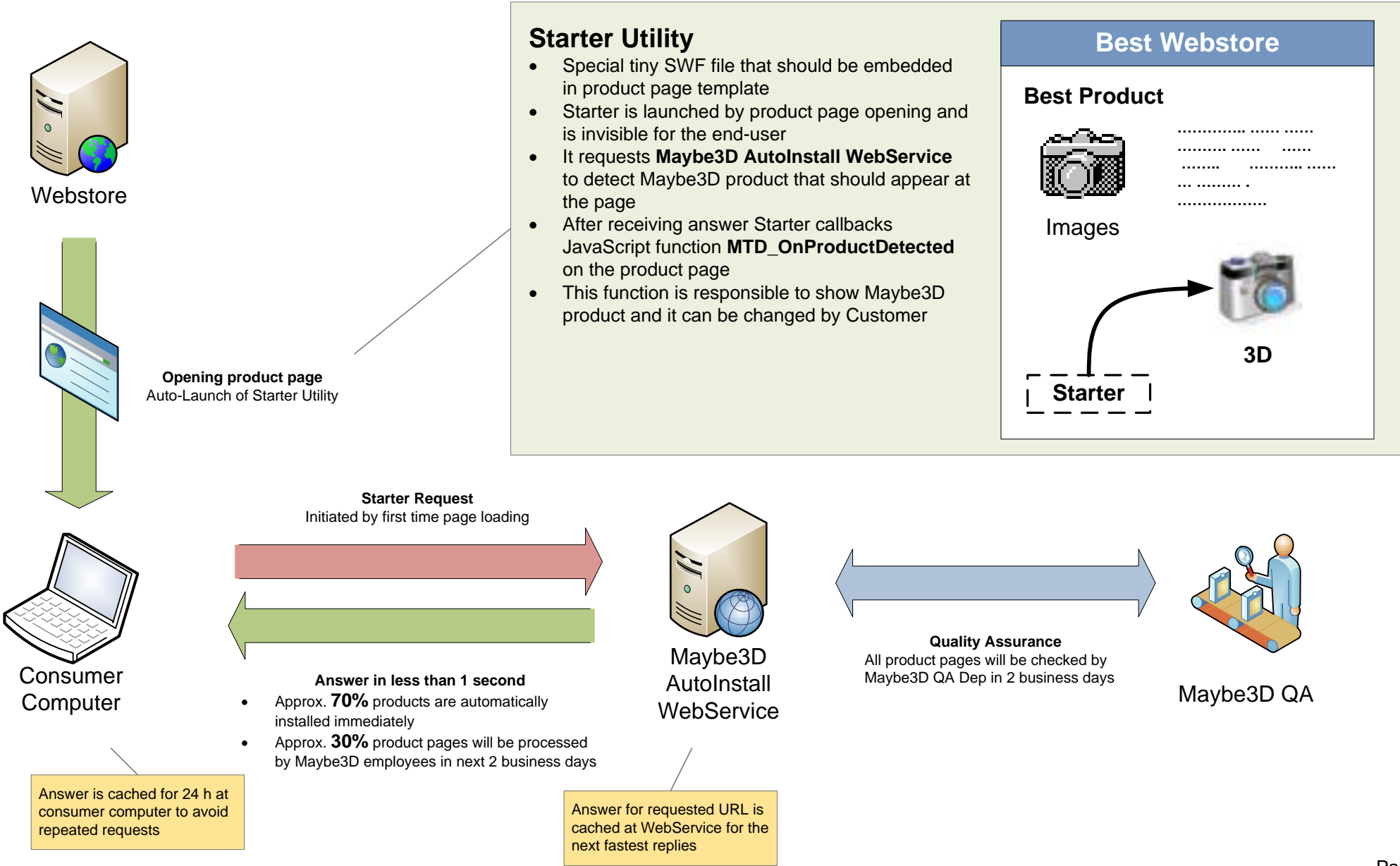
- Fast & Easy Installation. Significant human resources saving.
- No need to modify webstore server source code
- Every new page load emits request to AutoInstall Webservice (small lack of load 3D Product time & dependence of Maybe3D Server Status\*)

### Using Server-to-Server Request

- Need to modify webstore server source code
- Only few first time page loads emit requests to AutoInstall Webservice (fastest 3D Product loading & independence of Maybe3D Server Status\*)

\* Maybe3D Server with AutoInstall Webservice is available 24x7 excluding very rare short maintenance pauses.

# Automatic Installation Using Starter Utility



# Starter Utility Manual

---

## Setup

The products you need should be copied to a folder on the server, for example **mywebstore.com/3dmodels**. It is recommended to download all the models from the Maybe3D FTP server. This will eliminate the need to find out what exact models are needed and save time. This work for you will be done by Maybe3D server.

Download the archive with the starter utility:  
[http://services.maybe3d.com/files/MTD\\_Starter.zip](http://services.maybe3d.com/files/MTD_Starter.zip)

Unpack the archive to the server, for example to the folder **mywebstore.com/MTD\_Starter**.

Folder **MTD\_Starter** contains four files:

- **MTD\_Starter.swf** – starter utility, runs the needed product
- **swfobject.js** – library SWFObject 2 is used for easement of embedding starter and product
- **expressInstall.swf** – Adobe utility, that allows to upgrade Flash Player to the latest version
- **example.html** – example file shows the functions of the starter

## System Check

Open **example.html** in edit code mode.

Change the value of the variable **productFolder** at line 15 of block `<script type="text/javascript">` to the path to the models Maybe3D on your server. Do not include domain name.

For example, if the models are placed at **mywebstore.com/3dmodels**, you must specify the value as **productFolder: "3dmodels"**,

Run the example using path to the file **example.html**:  
**http://mywebstore.com/MTD\_Starter/example.html**

A page with an example should appear. In the bottom of the page there are sample content of product page, which will be used to determine product by Maybe3D Server. In the upper left corner you can see the starter.

Examine the output of the starter. Status should be **Pending**. This means that the server has started to determine which product should be installed. If you update the page, the product should appear in the upper right corner of the page. For the next page reloads the product appears almost immediately, because the result of the product determination is cached on server and user PC.

## Troubleshooting

If the product does not appear after restarting the page, you must pay attention to the debug log of the starter.

Look at the value of **Status**:

- **Pending** – the product is still determining (it may required the help of human on Maybe3D side)
- **InvalidUrl** – the server cannot access a page that displays the starter
- **UnknownProduct** – the product was not released by Maybe3D
- **Resolved** – this value with an invisible product means that SWF file of the product was not found. Look what value of **productPath** at the starter log is printed.

## Embedding product to the product page template

Embedding the starter into the template of product page should go the same way. It is recommended to embed it into the bottom of the page.

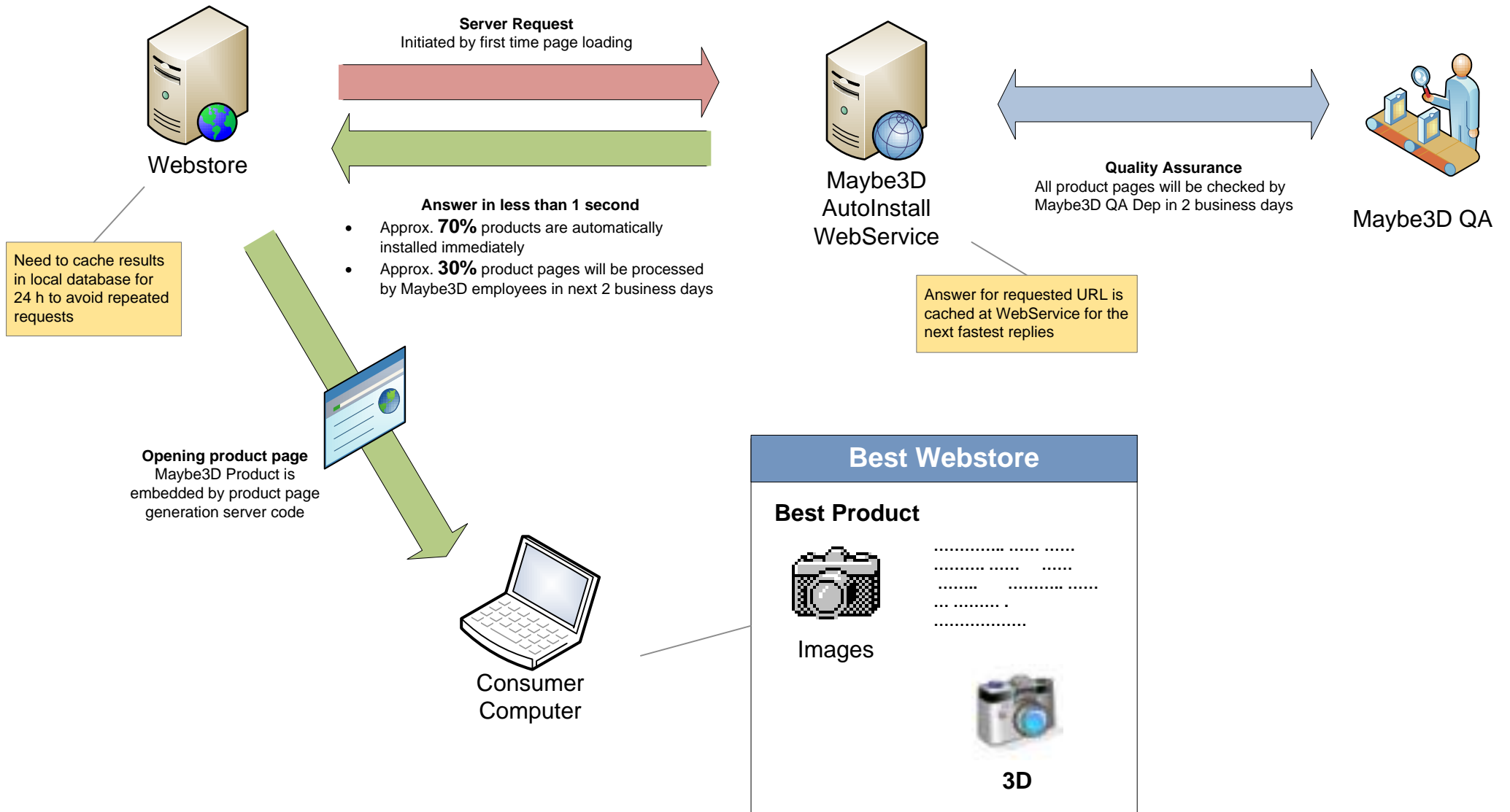
For testing you can use the code from the example. After all the checks you can set a small application size and set **debugMode: "false"**, so the end-user will not get a sight of the starter.

After determination of the product, the starter calls **JavaScript** function **MTD\_OnProductDetected(productPath)**, which should be contained in the code of the product page.

To implement this function you can use the same logic as in the **example.html** file to show 3D-model directly in the page.

Another way of this function use is to create a «View in 3D» button. When user clicks it, a popup window with 3D-model should appear.

# Automatic Installation Using Server-to-Server Request



# Server-to-Server Request Manual

To obtain the data, use the request to XML Web Service.

Actual address: <http://services.maybe3d.com/ai.asmx>

You must request function **DetectProduct**.

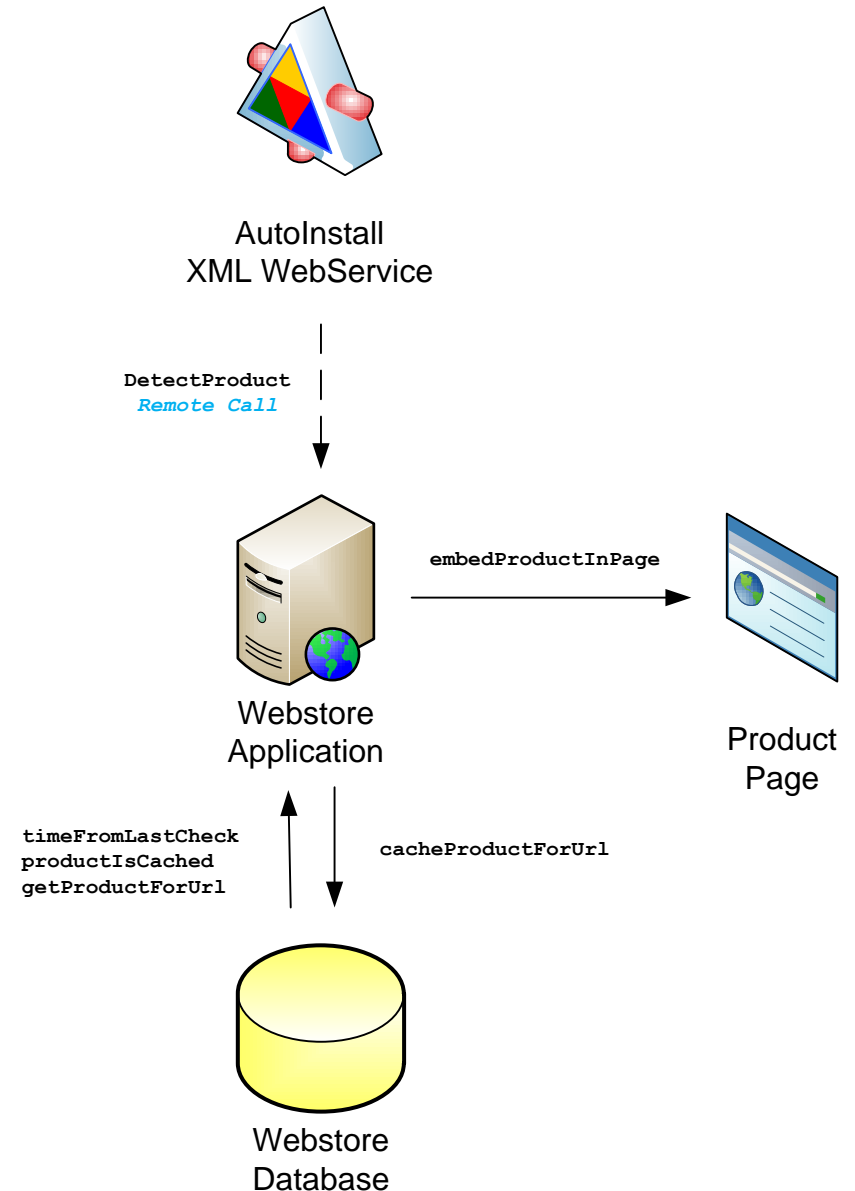
Returned result consists of 3 string type fields:

- **Status** – status of the determination of the product. Possible values: **Pending**, **Resolved**, **UnkownProduct**, **InvalidUrl**
- **MTD\_ID** – product identification number. Using of this field is optional
- **MTD\_FILENAME** – the name of SWF file of the product. For example, **Canon\_\_EOS\_60D**

The logic of calling the function can be described by the pseudo-code:

```
function onProductPageLoad(url) :  
    if timeFromLastCheck(url) < 24h and productIsCached(url) :  
        embedProductInPage(url, getProductForUrl(url))  
    else:  
        serviceResult = remoteCall_DetectProduct(url)  
  
        switch serviceResult.status:  
            case "Resolved":  
                cacheProductForUrl(url, serviceResult.MTD_FILENAME)  
                embedProductInPage(url, serviceResult.MTD_FILENAME)  
            case "InvalidUrl":  
                throw AIException("Service can not access to web page: " + url)  
            case "UnknownProduct":  
                cacheProductForUrl(url, null)
```

You can download an archive with the examples of the requests to the AutoInstall service on popular platforms .NET and PHP by the following address: [http://services.maybe3d.com/files/MTD\\_S-to-S.zip](http://services.maybe3d.com/files/MTD_S-to-S.zip)



# Privacy Policy

---

We understand the importance of maintaining confidentiality of your information.

The only information that is transferred from your web store to our server is a request for the definition of the product which must be installed on the page.

Our company uses special management and technical procedures to ensure that any data of your store is not going to anyone besides the responsible employees.

We also guarantee that the information received from your web store will never and under no circumstances be provided to third parties.

