

By Matthijs Alles - (c)2009-2011 Bixie
Adapted from the original VirtueUpload 1.0 from Tobias Kuhn.
License: GNU/GPL

For Joomla! 1.5 and VirtueMart 1.1.3 to 1.1.7

More info and support: www.bixie.org/forum

INSTALLATION:

Install the component `com_virtueupload.zip` and the module `mod_virtueupload.zip` in the backend of Joomla (Extensions->Install/Uninstall).

COMPONENT SETTINGS:

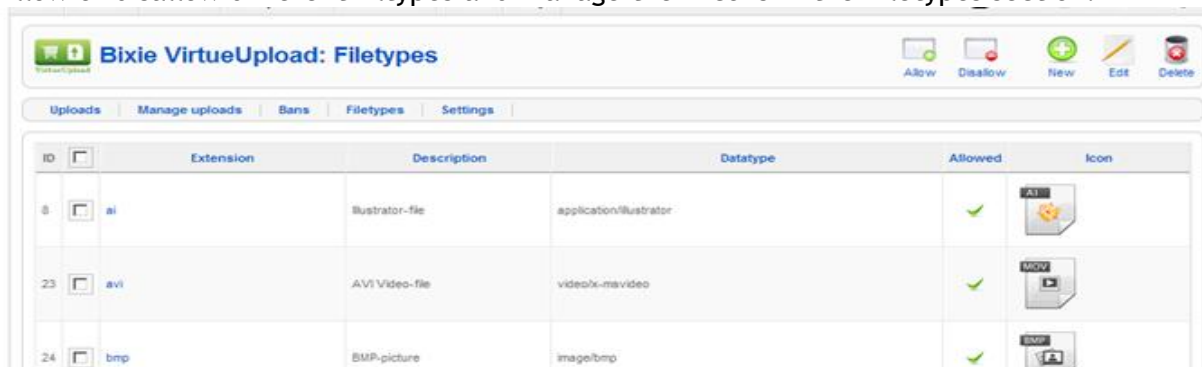
Default settings are:

Setting:	Default:	Description:
Upload folder	virtueupload	Default folder is created on component-installation. Make sure it's writable!
Theme	default	Select the theme. Upload new themes in the folder 'components/com_virtueupload/themes'
Number allowed uploads	5	The number of files that can be uploaded with one product. You can lower this number in the module-settings.
Show commentfield	Yes	Show the possibility to add a comment with an upload
Maximum length filename	0	The filename is cut off after the number of characters set. Value 0 uses complete filename.
Max. uploadsize	8024000	Maximum uploadsize in bytes. Remember to change your PHP-settings accordingly.
Show thumbnail	Yes	Shows a thumbnail or icon with the upload-information and mail.
Maximum size for thumbnail	8388608	The maximum size of the image in bytes for creating an thumbnail (max = 8Mb). The file-icon will be shown for bigger images.
Icon width	64px	Width of the icon.
Icon height	64px	Height of the icon.
Thumbnail width	125px	Width of the thumbnail.
Thumbnail height	125px	Height of the thumbnail.
Small thumbnail heigth & width	45px	Size of thumbnail in cart, order info and administrator

Setting:	Default:	Description:
VM Product attribute	Upload	Name of the attribute added to the VirtueMart-product. Case-sensitive!
Size inputfield	10	The size of the 'Upload'-attribute field in VirtueMart (where 'None' of the upload-id is printed)
Upload required		Require upload before ordering the product. You can override this setting in the module-settings.
Use ajax-addtocart	Yes	Switch this function on if VirtueMart uses ajax to update the cart.
Mail uploads to admin	Never	Mail a message with attachment to the administrators with every upload, of only at ordering of the product.
Maximum size attachment	4194304	The maximum size in bytes of the email-attachment.
Ban IP-addresses	0	The number of tries allowed before an IP address is banned. 0 switches it off.
Days old for 'Manage uploads'	7	Number of days an upload has to be old to appear and be deleted via 'Manage uploads'.
Copy file on FTP-server	No	Copy all uploads on a remote FTP-server. ATTENTION! Uploads deleted by user or admin are not deleted on the remote server!
FTP-serveraddress	ftp.yourserver.com	FTP-address of the remote FTP-server
FTP-username	user	Username for the remote FTP-server.
FTP-Password	pass	Password for the remote FTP-server
Folder on the remote FTP-server	0	Exact location of the files on the remote server. Make sure the folder is writable.
Leave copy on webserver	Yes	Leave copy of file on webserver. The thumbnail stays stored on the webserver.

FILETYPES:

Allow or disallow different filtypes and manage their icons in the Filetypes section.



MODULE SETTINGS:

Set module position to “upload”. You can type this value in the pulldown-menu (yes, you can!)

Details

Module Type:	mod_virtueupload
Title:	VirtueUpload Module
Show Title:	<input type="radio"/> No <input checked="" type="radio"/> Yes
Enabled:	<input type="radio"/> No <input checked="" type="radio"/> Yes
Position:	upload
Order:	0::VirtueUpload Module
Access Level:	Public Registered Special
ID:	23
Description:	<p>Module voor VirtueUpload. Te plaatsen in de flypage. Include with {loadposition positionname} in product description.</p>

Menu Assignment

Menus:	<input checked="" type="radio"/> All <input type="radio"/> None <input type="radio"/> Select Menu Item(s) from the List
--------	---

You can use several VirtueUpload-modules in different positions, just make sure to load the right position with {loadposition } in your product description.

By setting the module parameters, you can set different parameters for separate modules. This way you can vary the allowed uploads or required uploads for each module.

Parameters

▼ Module Parameters

Module Class Suffix	
Number allowed uploads	5
Upload required	<input type="radio"/> Default <input checked="" type="radio"/> Yes <input type="radio"/> No
Text above form	<h1>Upload your file
Show uploadinfo	<input checked="" type="radio"/> Yes <input type="radio"/> No
Cols comment-field	40
Rows comment-field	2

VIRTUEMART SETTINGS:

These settings are done in the backend-section of the VirtueMart component. Add these settings to each product you want to attach an upload to.

In **product edit**:

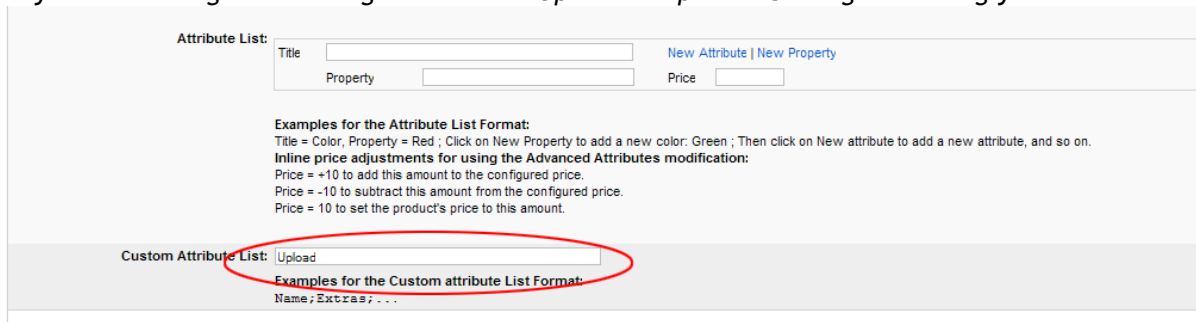
-Add to the **Product Description** the following tag anywhere in your text:

```
{loadposition upload}
```

If the module is published in a different position than 'upload', change it here.

-Add an "Custom Attribute" with the name "Upload".

This is done in 'Custom Attribute List' on the tab 'Product status'. You can use a different name, but you must change the setting in the VirtueUpload component Settings accordingly.



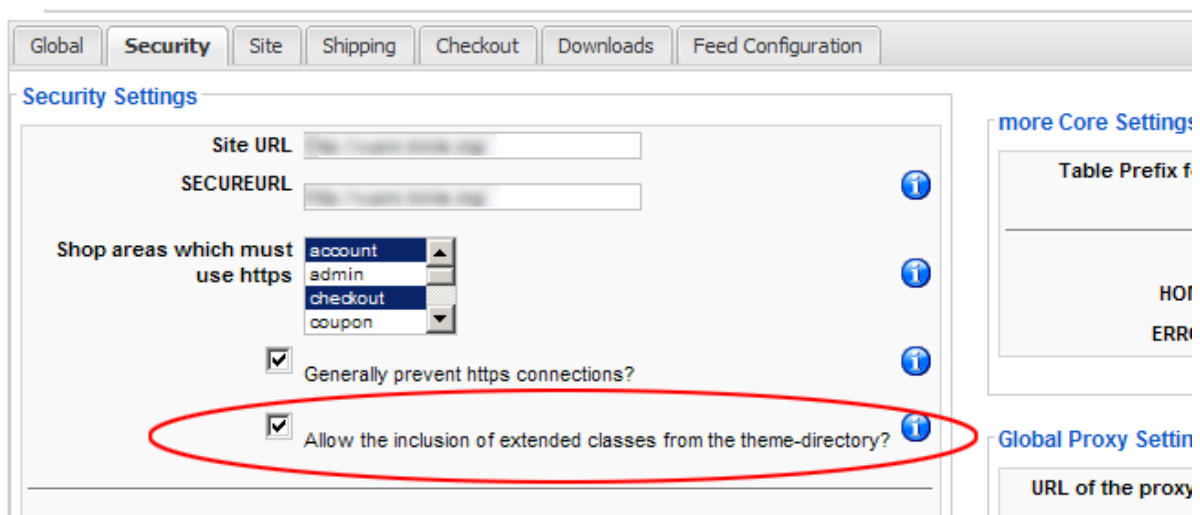
VirtueMart 1.1.4 ([Check for latest version](#))

Attention:

-Module mod_virtueupload must be published in position "upload".

-The option **Enable content mambots / plugins in descriptions?** in the Global VirtueMart config must be checked.

-From VirtueMart 1.1.5 VirtueUpload uses the user classes. This means that there are no core-files of VirtueMart are changed. All changes are made in the template directory. This way the changes are preserved when you update VirtueMart. This means you have to allow the user classes. This is done in the "security" tab of the global VirtueMart configuration.



VIRTUEMART PATCHES:

Installing patch

Go to the VirtueMart backend, to Admin menu option “Check for updates”. Then choose the second tab “Upload a patch”. Click browse and select the right patch for your VirtueMart version. Click “Upload & Preview” to proceed.

This is essentially an update patch for VirtueMart. Make sure you use the right files for your VirtueMart version! A list of files is shown. Click “Apply patch now” to overwrite your files.

Attention:

-There are separate patches for VirtueMart 1.1.3 to VirtueMart 1.1.6! Make sure you use the right ones!

-This only works on a standard VirtueMart 1.1.3 to 1.1.6 installation in default-theme. If you use an customized VirtueMart or custom theme, use the manual adjustments described below.

-Always make a backup before overwriting your files!

-For VirtueMart 1.1.4 the [Make custom attributes optional-hack](#) must be applied. The adjusted ps_cart file is included in the patch.

Manual adjustments

These adjustments do not need to be made when you installed the VirtueMart patch as described above.

Five files must be changed:

-In folder components/com_virtuemart/themes/yourthemename/templates

- /basket/basket_b2c.html.php
- /basket/ro_basket_b2c.html.php
- /pages/account.order_details.php
- /order_emails/confirmation_email.tpl.php

-In folder administrator/components

- /com_virtuemart/classes/ps_checkout.php

In VirtueMart 1.1.4, make sure you apply the [Make custom attributes optional-hack!](#)

Place on top of each file that you change, just below _JEXEC check:

```
if (file_exists
(JPATH_ADMINISTRATOR.DS."components".DS."com_virtueupload".DS."classes".DS."output.class.php")) {
require_once (JPATH_ADMINISTRATOR.DS."components".DS."com_virtueupload".DS."classes".DS."output.class.php");
}
```

Templatefiles

- In *basket/basket_b2c.html.php* (line 33) and *basket/ro_basket_b2c.html.php* (line 36) (same for vm1.1.3 and vm1.1.4):
Replace line:

```
<td><?php echo $product['product_name'] . $product['product_attributes' ]?></td>
```

With:

```
<td><?php echo $product['product_name'].$product['product_attributes'].  
VUOutput::UploadInfo ( $product['product_name'] , 'cart' ) ?></td>
```

- In *pages/account.order_details.php*
- Starting from line 361 you see (same for vm1.1.3 and vm1.1.4)::

```
// ***** Change guard to use new flag $link_to_product instead of !empty( $product_id )  
        if( $link_to_product) {  
            echo '<a href="'. $sess->url(  
$mm_action_url."index.php?page=shop.product_details&product_id=$product_id") .'" title="'. $dbcart->  
>f("order_item_name").'">';  
        }  
        $dbcart->p("order_item_name");  
        echo " <div style=\"font-size:smaller;\">" . $dbcart->  
>f("product_attribute") . "</div>";  
// ***** Change guard to use new flag $link_to_product instead of !empty( $product_id )  
        if( $link_to_product) {  
            echo "</a>";  
        }  
    }
```

Replace that bit with:

```
// ***** Change guard to use new flag $link_to_product instead of !empty( $product_id )  
        if( $link_to_product) {  
            $enda = "</a>";  
        }  
        if( $link_to_product) {  
echo '<a href="'. $sess->url( $mm_action_url."index.php?page=shop.product_details&product_id=$product_id") .'  
" title="'. $dbcart->f("order_item_name").'">';  
        }  
        $dbcart->p("order_item_name");  
echo $enda." <div style=\"font-size:smaller;\">" . $dbcart->f("product_attribute").  
VUOutput::UploadInfo ( $dbcart->f("product_attribute") , 'cart' ) . "</div>";  
// ***** Change guard to use new flag $link_to_product instead of !empty( $product_id )
```

- In `order_emails/confirmation_email.tpl.php` regel 184 (same for vm1.1.3 and vm1.1.4). Replace line:

```
<td><?php $dboi->p("product_name")?>  
<?php echo ($dboi->f("product_attribute") ? ' ('.$dboi->f("product_attribute").')' : ''); ?></td>
```

With:

```
<td><?php $dboi->p("product_name")?>  
<?php echo ($dboi->f("product_attribute") ? ' ('.$dboi->f("product_attribute").')' : ''); ?>  
<?php echo VUOutput::UploadInfo ( $dboi->f("product_attribute") , 'mail' ); ?></td>
```

VirtueMart Core-bestanden

To be able to add an order ID to the upload when the product is ordered, you must hack one VirtueMart core file. Remember that this will be lost when you update VirtueMart.

- In `administrator/components/com_virtuemart/classes/ps_checkout.php`, line 1101 (same for vm1.1.3 and vm1.1.4):
Just below this line:

```
$description .= $ps_product->getDescriptionWithTax($_SESSION['cart'][$i]["description"],  
$dboi->f('product_id'));
```

Add this line:

```
VUOutput::AddOrderid( $description , $order_id );
```