

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");
}
```

Template files

- 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 );
```