

# Tutorial Virtuemart Themes

---

Mit der Version 1.1 wurden in Virtuemart die sogenannten „Themes“ eingeführt. Damit ist es endlich möglich, fast alle Ausgaben in Virtuemart nach eigenen Wünschen anzupassen, ohne die Originalprogramme verändern zu müssen.

Mit diesen Themes können Grafiken, HTML-Ausgaben und PHP-Funktionalität verändert werden. Anders als bei bekannten „echten“ Template-Systemen wie patTemplate oder Smarty gibt es hier keine echte Trennung von Layout und Code. Die Themes sind PHP-Dateien, in denen HTML und PHP-Logik gemischt enthalten sind.

Dieses Tutorial soll einen kleinen Einblick in den Aufbau der Themes und die Möglichkeiten verschaffen. Eine komplette Anleitung zur Programmierung kann hier – schon aufgrund der Komplexität von Virtuemart – nicht vermittelt werden.

## Bevor wir anfangen

### Grundsätzliche Überlegungen

Bei jeder Internetpräsenz und ganz besonders bei einem Onlineshop sollte man sich bestimmte Dinge überlegen, bevor man anfängt, sein Layout und seine Funktionen zu planen. Die wichtigste Überlegung ist die „anvisierte Zielgruppe“. Daraus resultieren eine Reihe von Rahmenbedingungen, die bei der Konzeption der Seite eine Rolle spielen (sollten).

Bei einer technisch versierten Zielgruppe (z.B. Gamer) kann der Betreiber der Seite von ganz anderen Erwartungen und technischen Voraussetzungen ausgehen als bei anderen Zielgruppen.

Somit sollte der Einsatz von Funktionen, die eine spezielle Technologie voraussetzen (Flash, AJAX, ...) genau überlegt werden und ggf. Ersatzfunktionalität integriert werden. Virtuemart tut dies beispielsweise vorbildlich: wenn (aus welchen Gründen auch immer) nicht die Möglichkeit besteht, den Artikel per AJAX zum Warenkorb hinzuzufügen, wird (wie früher) die Warenkorbseite aufgerufen und so der Artikel hinzugefügt.

### Gedanken zu Sicherheit und Browserkompatibilität

Da die Virtuemart Themes eine Mischung aus HTML und PHP sind und einige wichtige Funktionen in die Themes eingebettet sind, sollte ein gewisses Know-How bei der Überarbeitung dieser Dateien vorhanden sein. Jedes Virtuemart-Release wird intensiv getestet und kann als relativ sicher und fehlerfrei betrachtet werden. Wenn jetzt Eingriffe vorgenommen werden, kann dieses „Kartenhaus“ durch eine unbedachte Änderung einstürzen.

Jeder Surfer stellt an ein Ecommerce-System natürlich höhere Anforderungen an Fehlerfreiheit und Sicherheit als an z.B. eine private Homepage. Ein Kunde, der in Ihrem Shop merkwürdige Fehlermeldungen und unerklärliche Reaktionen feststellt wird sicherlich nicht kaufen und vielleicht auch nicht noch einmal wiederkommen. Daher sollten Sie bei Änderungen am System das notwendige Know-How besitzen oder Fachleute beauftragen. Auf jeden Fall sind Tests der geänderten Dateien unter verschiedenen Voraussetzungen zwingend notwendig!

Weiterhin kann man sich darauf verlassen, dass die von Virtuemart gelieferten Themes in allen wichtigen Browsern und Betriebssystemen korrekt angezeigt werden. Auch dies kann bei individuellen Änderungen schnell nicht mehr der Fall sein. Ein Test der Änderungen in verschiedenen Systemumgebungen ist unbedingt notwendig.

## Jetzt geht es los

### Wie finde ich die Themes?

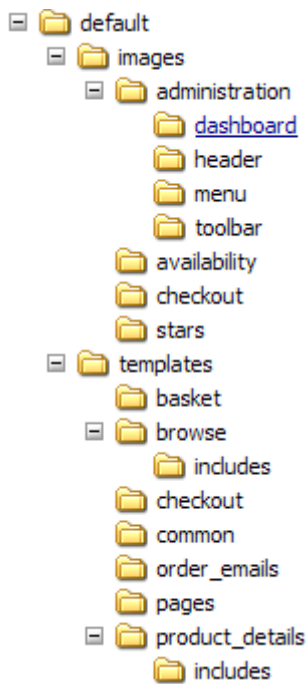
Die Themes liegen in [Joomla-Root]/components/com\_virtuemart/themes/. Dort befindet sich nach der Installation das "Default-Theme". Davon sollte man sich als erstes eine Kopie anlegen und nach eigenem Wunsch benennen. Das sieht dann ungefähr so aus: [Joomla-Root]/components/com\_virtuemart/themes/mytheme/. Danach wird in der Virtuemart-Administration dieses neue Theme aktiviert:

The screenshot shows the Virtuemart administration interface. On the left is a sidebar menu with 'Administration' expanded and 'Konfiguration' selected. The main content area has tabs for 'Global', 'Security', 'Seite', 'Lieferung', 'Bestellung', 'Downloads', and 'Feed Configuration'. The 'Seite' tab is active, showing a 'Display' section with various checkboxes and dropdowns. The 'Layout' section is highlighted in light blue and contains the following settings:

- Wählen Sie das Thema/Stil für den Shop: default (dropdown menu with options: default, ja-mesolite, vm\_white)
- Produkte pro Zeile: 1
- Standard-Kategorie-Vorlage: managed
- FLYPAGE: flypage.tpl
- Dynamisches Verkleinern von Produktbildern aktivieren?:
- Breite des verkleinerten Bildes: 90
- Höhe des verkleinerten Bildes: 90

Zum leichteren Auffinden der einzelnen Theme-Elemente sollte man in seiner Joomla-Installation SEO-Erweiterungen deaktivieren, da in den Virtuemart URLs die Namen der Elemente gut abzulesen ist. Wenn diese URLs durch SEO-Tools nicht zu sehen sind, erschwert man sich die Einarbeitung in die Theme-Struktur.

## Theme Struktur



Das Themes-Verzeichnis enthält über 340 Dateien in 19 Verzeichnissen. Dies liegt daran, dass die allermeisten Darstellungselemente von Virtuemart als einzelne Template-Dateien vorliegen. Das erschwert zu Anfang etwas die Einarbeitung, hat aber erhebliche Vorteile.

Im Hauptverzeichnis des Theme liegen einige wichtige Dateien, die für die Konfiguration und die grundlegende Funktionalität des Templates wichtig sind.

Die Verzeichnisstruktur der Themes ist gut und einfach nachvollziehbar gegliedert. Zuerst erfolgt die Trennung nach „Images“ und „Templates“, darunter die Gruppierung nach den einzelnen Funktions- und Seitenbereichen.

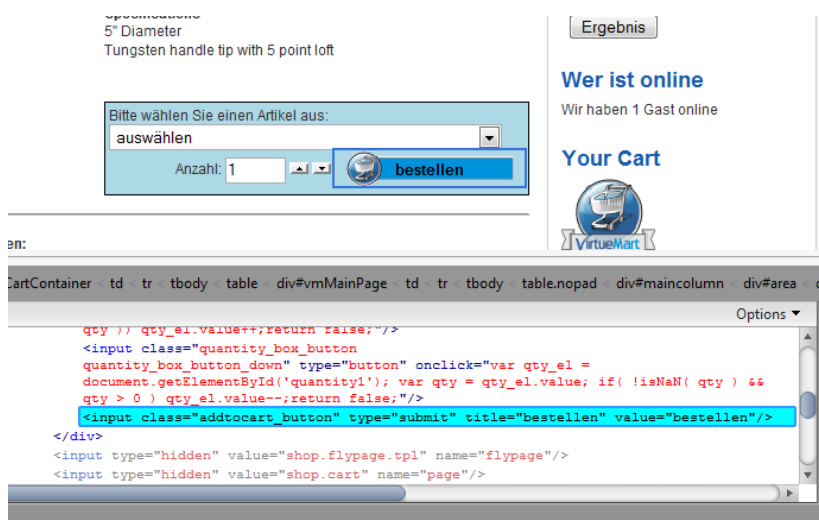
Das Images-Verzeichnis enthält alle grafischen Elemente (Buttons usw.) des Themes und ist relativ einfach zu verstehen. Durch die Verzeichnisstruktur und nach einer Durchsicht der Grafiken sollte klar sein, wo diese zur Anwendung kommen. Diese nehmen mit 255 den größten Teil der Themes-Dateien ein. Die Grafiken können einfach durch eigene Dateien ersetzt werden. Natürlich sollte man peinlich genau auf die Abmessungen (Breite und Höhe) der Originale achten, damit nach dem Austauschen der Bilder die Layouts noch funktionieren. Natürlich können auch abweichende Dimensionen genutzt werden, dann ist in der Regel auch eine Überarbeitung des jeweiligen Templates notwendig.

Das Templates-Verzeichnis enthält die bereits angesprochenen PHP-Dateien zur Steuerung der Ausgaben. Auch hier erleichtert eine „sprechende“ Verzeichnisstruktur die Einarbeitung. Dennoch ist es hier etwas komplexer, die gewünschte Datei zu ermitteln.

Dazu kommt die bereits angesprochene Virtuemart-URL ins Spiel. Hier eine typische „Product Flypage“ (Produkt-Detailansicht):

```
/index.php?page=shop.product_details&category_id=11&flypage=flypage.tpl&product_id=37&option=com_virtuemart&Itemid=71&vmcchk=1&Itemid=71
```

Die beiden hervorgehobenen Elemente der URL erleichtern uns das Auffinden der benötigten Datei: im Verzeichnis Templates liegt das Unterverzeichnis „product\_details“ und darin die Datei „flypage.tpl“ – und schon haben wir die Datei gefunden. Das funktioniert bei den meisten Templates ganz gut. Bei einigen Templates ist es nicht ganz so einfach – insbesondere beim Warenkorb, den Checkout-Seiten und den Common- und Includes-Elementen (diese repräsentieren keine „komplette“ Seite, sondern sind einzelne Seitenelemente – wie z.B. der „Add-To-Cart-Button“). Aber auch hier sollte man mit ein wenig Virtuemart-Routine nach einer kurzen Einarbeitung zum Ziel kommen. Alle Dateien haben (zumindest im Virtuemart-Kontext) selbsterklärende Dateinamen.



In ganz hartnäckigen Fällen der Unauffindbarkeit hilft ein Blick in den Seitenquelltext (sehr zu empfehlendes Hilfsmittel: Firebug-Extension für Firefox) und eine Volltextsuche nach markanten Passagen aus dem Quelltext über den Templates Verzeichnisbaum.

## Programmierung der Templates

### Technischer Hintergrund / Aufbau der Template-Dateien

Es ist auch hier wichtig zu wissen, dass die Virtuemart Template-Dateien nicht mit „echten“ Template-Systemen vergleichbar sind. Hier gibt es keine gekapselte Template-Logik mit Platzhaltern usw. Die Template-Dateien sind vielmehr mit typischen PHP-Includes vergleichbar. An einer bestimmten Stelle im Code werden diese Dateien hinzugeladen und **ausgeführt** (!). Das bedeutet, dass ein Fehler in der Template-Datei unweigerlich zu einem Fehler in Virtuemart wird.

Ein weiterer Aspekt ist, dass bei der Ausführung der Templates der „Variablen-Kontext“ des ausführenden Programms – und zwar genau an der Stelle der Ausführung – zur Verfügung steht. Ein Nachteil bei „gewachsenen“ Systemen (wie Virtuemart) ist häufig die uneinheitliche Programmierung der einzelnen Funktionen. Dies ist leider auch hier der Fall und erschwert gerade im Bereich der Nutzung der Variablen die Einarbeitung. Unterschiedliche Namenskonventionen und Benennungen und eine von Joomla abweichende Datenbankklasse müssen hier berücksichtigt werden. Oft hilft hier nur ein Dump der Variablen (z.B. PHP-Funktion `print_r()`) direkt im Template, um an die zur Verfügung stehenden Variablen zu kommen. Bei weitergehenden Template-Anpassungen wird ein Studium der Virtuemart-Funktionen und Klassen unerlässlich sein. Hier hilft ein spezieller PHP-Editor, der die Verknüpfungen selber auflösen kann und somit das Auffinden der Funktionen erleichtert.

Der Code im Template wird sofort ausgeführt. Ein Echo-Befehl wird sofort ausgegeben und eine Änderung der von Variableninhalten bewirkt, dass der neue Inhalt von allen nachfolgenden Programmfunktionen verarbeitet wird.

Dieses Verhalten macht das Template-System extrem flexibel, da nicht nur Layout-Änderungen möglich sind, sondern auch eigene Funktionen eingebaut werden können. Dadurch können aber auch sehr schnell Fehler oder Sicherheitslücken in ein eigentlich als sicher geltendes Virtuemart eingebaut werden. Eine gewisse Sachkunde sollte bei der Bearbeitung also vorhanden sein.

Jede Datei beginnt (wie bei Joomla üblich) mit der Prüfung, ob die Datei innerhalb von Joomla geladen wurde. Dies sollte unbedingt auch bei eigenen Templates aus Sicherheitsgründen beibehalten werden.

Danach folgt unmittelbar der auszuführende Code des Templates. Wie bereits erwähnt, wird dieser Code in die Programmausführung von Virtuemart direkt eingebunden und kommt sofort zur Ausführung. Da dieser Code stark von dem Virtuemart-Bereich abhängig ist, kann hier kein allgemeingültiger Überblick über die Programmierung gegeben werden. Mit HTML- und PHP-Kenntnissen sollte man aber schnell die Logik des einzelnen Templates verstehen können (ohne diese Kenntnisse sollte man sowieso lieber keine Änderungen vornehmen...)

## Spezielle Dateien

### Theme.php

Im Hauptverzeichnis des Themes liegt die „theme.php“. In dieser Datei erfolgt die Definition der Klasse „vmTheme“ (abgeleitet von „vmTemplate“). Durch den Zugriff auf diese Klassendefinition kann der Programmierer dem Theme eigene „globale“ Funktionen zur Verfügung stellen, die im gesamten Kontext des Themes zur Verfügung stehen.

Ein Zugriff auf die Funktionen innerhalb der Templates erfolgt über folgenden Aufruf:

z.B.: `<?php echo $this->vmlistAdditionalImages( $product_id, $images ) ?>`

wobei „\$this“ der Verweis auf das Theme-Objekt ist.

### Theme.xml

Ebenfalls im Theme-Hauptverzeichnis liegt diese Datei. Ähnlich wie bei Joomla-Extensions können hier Parameter definiert werden, die über die Theme-Konfiguration im Virtuemart Backend konfiguriert werden können. Dabei kommt die von Joomla bekannte Element- und Parameterstruktur zur Anwendung. Beispiel:

```
<param name="MyVariableName" type="radio" default="1" label="MyFieldTitle"
description="My Field Description">
  <option value="1">Yes</option>
  <option value="0">No</option>
</param>
```

Mit diesem Beispiel stellen wir dem Theme die Variable „MyVariableName“ zur Verfügung. In der Administration sehen wir zwei Radiobuttons mit der Auswahl „Yes“ und „No“, die den Wert „1“ für Yes und „0“ für No abspeichern. Im Template kann mit folgendem Befehl auf diese Variablen zugegriffen werden: `$this->get_cfg('MyVariableName')`. Codebeispiel:

```
If ($this->get_cfg('MyVariableName')) {
  echo 'In der Administration wurde YES gespeichert';
} else {
  echo 'In der Administration wurde NO gespeichert';
}
```

### theme.config.php

Die in der theme.xml definierten und der Administration gespeicherten Parameter werden in dieser Datei (auch im Theme-Hauptverzeichnis) abgelegt. Dies ist eine reine Textdatei, in der im Klartext die Parameter und deren Werte enthalten sind. Diese Datei kann also auch einfach im Texteditor bearbeitet werden. Das obige

Beispiel wäre so in der Datei enthalten (für „no“ und „yes“):

```
MyVariableName=0  
MyVariableName=1
```

## Weitere Dateien

Ebenfalls im Hauptverzeichnis des Themes liegen noch .css und .js Dateien. Diese Stellen (wie zu erwarten) CSS-Klassen und Javascript-Funktionen zur Verfügung. Da hier jedoch keine Besonderheiten in Bezug auf Virtuemart zu beachten sind, können diese Dateien wie jede andere ihres Typs behandelt werden.

## Möglichkeiten und Grenzen

Durch die Themes ist es endlich möglich, weitreichende Änderungen an Virtuemart vorzunehmen, ohne den Quellcode zu verändern. Das dies in der Vergangenheit ein notwendiges Übel war, wird jeder bestätigen können, der nicht mit den von Virtuemart vorgegebenen Ausgaben leben konnte. Die besondere Problematik in diesem Zusammenhang ist die Updatefähigkeit. Wenn Virtuemart verändert wurde, kann ein Update natürlich nicht problemlos eingespielt werden. Daraus folgt: wenn der Aufwand für ein Update hoch ist, wird es nicht / erst später installiert. Dies ist natürlich gerade bei einem Onlineshop nicht zu verantworten. Durch die Themes wird ein großer Teil der sogenannten „Hacks“ überflüssig, womit Updates wieder einfach eingespielt werden können.

Durch die Vermischung von HTML und PHP (Layout und Funktion) hat der Programmierer die Möglichkeit, neben den Layoutänderungen auch eigene Funktionalität in die Templates zu integrieren. Auch dies hilft – bei guter Planung und Know-How – Hacks zu vermeiden.

Allerdings bleiben die Funktionserweiterungen auf den Zeitpunkt der Ausführung des Templates beschränkt. Grundlegende Änderungen an der Virtuemart Funktionalität sind somit nur sehr begrenzt möglich. Die in der Virtuemart Roadmap schon für die nächste Version geplanten Erweiterungen (z.B. Plugin-Fähigkeit) lassen aber hoffen, dass durch die Kombination „Themes und Plugins“ eine standardkonforme einfache Erweiterbarkeit von Virtuemart möglich wird.

## Tellerrand (Tipp am Rande)

Bei der Überarbeitung der Templates werden häufig zwei wichtige Punkte ausser Acht gelassen: die Sprachdateien und die Emailformulare. Beide sind jedoch nicht zu unterschätzen. Für eine zielgruppengerechte Ansprache sollten die Sprachdateien auf das „Wording“ Ihrer Zielgruppe angepasst werden (oder glauben Sie, jeder weiss was die „Account-Verwaltung“ ist?).

Gleiches gilt für die Email Templates. Geben Sie Ihrem Kunden auch nach dem Kauf das Gefühl, „das Richtige getan zu haben“. Ein individuelles Email-Layout mit Ihren speziellen Informationen (Serviceangebot, Ansprechpartner für Support, ...) steigert die Wahrscheinlichkeit auf Folgegeschäfte!