PHOSPHOR MEDIA
Your success is our business

Merchant Optimizer

Making Your Site Faster and More Search Friendly

Installation and Licensing Instructions

Installation and licensing procedures are covered in the Module Installation and Licensing Guide.

Setting Up Optimizer

We have invested a considerable amount of time making Optimizer easy to use. In fact, most can use Optimizer without making a single change to its settings. However, we STRONGLY encourage you to read through the following information to ensure you get the maximum benefit from Merchant Optimizer.

Optimizer will also let you preview the generated pages before the rest of the world will see them. We call this preview Staging Mode. Once you have reviewed the results, and made other preparations, you will switch Optimizer to Live Mode.

The overall tasks for using Optimizer for the first time are:

- ❑ Review Optimizer settings in the Settings tab.
- ❑ Create Optimizer pages in Staging Mode (the initial default).
- ❑ Review the pages created by Optimizer.
- ❑ Switch Optimizer to Live Mode.

Review Optimizer Settings

Optimizer detects and sets defaults based on several different settings in your store, so you will probably not need to change them. However, you should review them in case a detected setting is no longer used. (For example, if SEO Settings/Short Links On is left enabled even though you are not actually using it. In this case, you would want to turn OFF the Short Links setting so it doesn't interfere with Optimizer.)

To review the Optimizer default settings

1. In your store's admin, open the Utilities folder.
2. Click the Merchant Optimizer link.
3. If not selected, click the Settings tab.

Folder/File Names

Prefix Folder: This is an optional directory you can specify from which Optimizer files will run. For example, if you previously used a directory structure such as:

```
Mydomain.com/great-site/index.html
Mydomain.com/great-site/categories/category_code.html
Mydomain.com/great-site/products/product_code.html
```

In this instance, the prefix directory was used to add a more search relevant key word.

If your SEO Settings Specified a Prefix, that value will automatically be used for this setting.

Home Page Filename: The file name used for Home Page of your site (which is a copy of your Merchant Storefront Screen). There is usually no reason to change this.

Category Product Directory: The directory where category pages will be stored. In Staging Mode this will be a predefined subdirectory called “pmStage”. In Live Mode, this will be a subdirectory of your server’s root directory (where the Home Page resides) unless you are using a Prefix Folder. If you are using SEO Settings/Short Links, then this will be preset to match those settings.

Product Page Directory: The directory where product pages will be stored. In Staging Mode this will be a predefined subdirectory called “pmStage”. In Live Mode, this will be a subdirectory of your server’s root directory (where the Home Page resides) unless you are using a Prefix Folder. If you are using SEO Settings/Short Links, then this will be preset to match those settings.

Preferences

- **Admin Update Mode:** Choose:

  Automatically run Optimizer when updating product/category pages if you only make infrequent changes to products and categories through the Merchant Admin. **(RECOMMENDED)**

  Manually run Optimizer in the Product/Category tab OR when updating product/category pages when you make frequent changes to multiple categories and products. You can either let Optimizer update its pages as you update the item or update them all through the Optimizer interface after you are done.

  Note: There is an additional method of telling Optimizer which pages to update that you use when updating categories and products through an import file. This method is covered in “Updating Pages with an Import File”

Category Pagination: By default, Optimizer ignores pagination on category pages, creating a Category Page with ALL products in that category on one page.

Optimizer does this for several reasons:

- Because the resulting page is static, it’s not as big an issue to have 12, 24, or even 36 products per page.
- Search engines (and most humans too) would rather scroll than ‘click’.
- If you have more than 40 or 50 products per category, you are better off having better categorization…customers just don’t like look through huge lists of specific items.

If you still want to use pagination, no problem…the first page of the category will be created as an Optimizer page, however, when the customer clicks the “Next” button, they will be taken to a Miva Merchant generated page. To do this, uncheck the Ignore Pagination checkbox.

Page Count: The page count setting determines the number of pages Optimizer creates at a single time. The default works for most sites. If Optimizer is timing out when creating pages, decrease this number. You can try a higher number if you have a lot of pages to create as this will decrease the time it takes Optimizer to create pages.

Creating Your First Set of Optimizer Pages

After you have reviewed your settings, it’s time to create your first set of Optimizer pages. We recommend creating Merchant pages first (since there are fewer of them) and then making a brief inspection of them to ensure that they are being created correctly. After that, you can create your Product and Category pages.

After you’ve created a complete copy of your site, use the procedure in “Updating Your Merchant Optimizer Pages” to learn about keeping your Optimizer pages up to date.

Note: On large sites (3000+ products) creating Product/Category page can take anywhere from 20 minutes to several hours. You will need to keep the Optimizer page open during this time and not disturb or close it.

To create your first set of Optimizer Merchant pages

1. In the Optimizer Admin, click the Merchant Pages tab.
2. Click the Update button.

   Optimizer creates pages for recommended non product and category Merchant pages such as your Store Front (Home Page), About Us (abus.html), FAQs (faqs.html), Privacy Policy (prop.html) and Sitemap (smap.html). (If you have our Merchant Moogle module, that actual list will be bases on the Merchant pages listed there.)
3. When Optimizer is done, it will present a link to the newly created Optimizer Home Page. Click this link (it open in a new window) and review for appearance. (Note: Category and Product page links WILL NOT WORK as they haven’t been created yet.)

To create your first set of Optimizer Product and Category pages

1. In the Optimizer admin, click the Product/Category Pages tab.
2. Click Update.

   Optimizer reloads and starts processing all your product and category pages. While it does this, Optimizer will show you the progress along with an estimated time for completion. You can leave this page, but be careful not to CLOSE this window or you will have to start over.
3. When Optimizer is done, go back to the Merchant Pages tab, and click the link to your Optimizer Home Page. Click around (be sure to click only Optimizer created Merchant pages or product and category pages). Usually, you only need to check one Merchant page, one Category page and one Product page. If everything looks ok and links are correct in one, it will be correct for all others.
4. You are now ready to go Live.

Switching to Live Mode

Switching between from Staging Mode to Live Mode is easy. However, before you make the switch, you need to take a look around your site at a few other details. Look through the following descriptions and follow the instructions beneath the one that exactly describes your current site. If one of the following doesn’t match, please use the Support tab in the module to send us a request for assistance.

Before I Installed Optimizer:

My site used standard Merchant URLs. For example: www.mysite.com/mm5/merchant.mvc?Screen=PROD&Product_code=this-product.

- You are ready to go, just use the **To switch to Live Mode** procedure below.
- If you used a directory index rule in your .htaccess file, you need to change it out as follows:

  ```
  #DirectoryIndex /mm5/merchant.mvc?Screen=SFNT
  DirectoryIndex index.html
  ```

My site used Merchant's built in SEO Settings/Short Links. For example: www.mysite.com/prod/this-product.html

- After you use the **To switch to Live Mode** procedure below, you will need to edit your .htaccess file as follows:

ROM

```
### Begin - Inserted by Miva Merchant
DirectoryIndex /mm5/merchant.mvc?Screen=SFNT

RewriteEngine On
RewriteRule ^mm5/admin.mvc? - [QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^product/([^/.]+).html$ /mm5/merchant.mvc?Screen=PROD&Product_code=$1 [QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^category/([^/.]+).html$ /mm5/merchant.mvc?Screen=CTGY&Category_code=$1 [QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^([^/]+)/([^/.]+).html$ /mm5/merchant.mvc?Screen=PROD&Category_code=$1&Product_code=$2
[QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^([^/.]+).html$ /mm5/merchant.mvc?Screen=$1 [QSA,L]
### End - Inserted by Miva Merchant
```

TO

```
### Begin - Inserted by Miva Merchant
#DirectoryIndex /mm5/merchant.mvc?Screen=SFNT

#RewriteEngine On
#RewriteRule ^mm5/admin.mvc? - [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^product/([^/.]+).html$ /mm5/merchant.mvc?Screen=PROD&Product_code=$1 [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^category/([^/.]+).html$ /mm5/merchant.mvc?Screen=CTGY&Category_code=$1 [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^([^/]+)/([^/.]+).html$
/mm5/merchant.mvc?Screen=PROD&Category_code=$1&Product_code=$2 [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^([^/.]+).html$ /mm5/merchant.mvc?Screen=$1 [QSA,L]
### End - Inserted by Miva Merchant
```

To switch to Live Mode

1. In the Optimizer Admin, click the Merchant Pages tab.
2. Check the Change to Live Mode option, and then click Update.

Switching Back to Staging Mode

If you need to take your optimizer pages "off line" for a while—for example, you only use optimizer to handle heavy traffic cycles—you can easily set your site back to standard Merchant.

Before I Installed Optimizer:

My site used standard Merchant URLs. For example: www.mysite.com/mm5/merchant.mvc?Screen=PROD&Product_code=this-product.

□ If you used a directory index rule in your .htaccess file, you need to change it back as follows:

```
DirectoryIndex /mm5/merchant.mvc?Screen=SFNT
#DirectoryIndex index.html
```

□ Then, use the **To switch back to Staging Mode** procedure below.

My site used Merchant's built in SEO Settings/Short Links. For example: www.mysite.com/prod/this-product.html

□ Change your .htaccess file as follows:

FROM

```
### Begin - Inserted by Miva Merchant
#DirectoryIndex /mm5/merchant.mvc?Screen=SFNT

#RewriteEngine On
#RewriteRule ^mm5/admin.mvc? - [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^product/([^/.]+).html$ /mm5/merchant.mvc?Screen=PROD&Product_code=$1 [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^category/([^/.]+).html$ /mm5/merchant.mvc?Screen=CTGY&Category_code=$1 [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^([^/]+)/([^/.]+).html$
/mm5/merchant.mvc?Screen=PROD&Category_code=$1&Product_code=$2 [QSA,L]

#RewriteCond %{REQUEST_FILENAME} !-s

#RewriteRule ^([^/.]+).html$ /mm5/merchant.mvc?Screen=$1 [QSA,L]
### End - Inserted by Miva Merchant
```

TO

```
### Begin - Inserted by Miva Merchant
DirectoryIndex /mm5/merchant.mvc?Screen=SFNT

RewriteEngine On
RewriteRule ^mm5/admin.mvc? - [QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^product/([^/.]+).html$ /mm5/merchant.mvc?Screen=PROD&Product_code=$1 [QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^category/([^/.]+).html$ /mm5/merchant.mvc?Screen=CTGY&Category_code=$1 [QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^([^/]+)/([^/.]+).html$ /mm5/merchant.mvc?Screen=PROD&Category_code=$1&Product_code=$2
[QSA,L]

RewriteCond %{REQUEST_FILENAME} !-s

RewriteRule ^([^/.]+).html$ /mm5/merchant.mvc?Screen=$1 [QSA,L]
### End - Inserted by Miva Merchant
```

□ After you use the **To switch to Live Mode** procedure below, you will need to edit your .htaccess file as follows:

To switch to Staging Mode

1. In the Optimizer Admin, click the Merchant Pages tab.

2. Check the Reset to Staging Mode

Note: You will have to rebuild your optimizer pages before going live again.

Updating Optimizer Pages

Whenever you add new or edit existing products and categories, or add new content to the Merchant pages managed by Optimizer, you'll need to update their corresponding Optimizer page. Optimizer provides three basic methods of updating these pages.

Updating Products and Categories

For products and categories, the easiest way is to use the default "Automatically run Optimizer when updating product/category pages" option. When this option is selected, the corresponding product and category page is automatically updated whenever you add or edit them through the Merchant admin Products and Category admin screens.

If this option is NOT selected, Optimizer provides an checkbox option on those admin screens that when checked, will perform the updates when that screen is updated.

If neither of the proceeding options is used, you can still update product and category pages as follows:

To update product and category pages that have been added or edited

1. In the Optimizer admin, open the Product/Category pages tab.
2. Ensure that Changed Records is selected.
3. Click update.

If you have created or edited products or categories via an Import file, you should use the Import update feature in Optimizer. You will need a copy of the import file you used to do the update, in the following format:

Product Code|Product Name

Or

Category Code|Category Name

IMPORTANT: Make sure that the delimiter is a pipe character ('|').

To update imported product and category pages

1. In the Optimizer admin, open the Import tab.
2. Open the import file you prepared previously.
3. Copy the entire contents of that file to your computers clipboard
4. In the Import tab, paste the clipboard contents in the text box provided.
5. Select whether or not you are importing Product or Categories.
6. Click Update.
7. When Optimizer finishes, it will populate the relative Changed records list under the Product/Category Pages tab. You will then use the standard method to update those changed records.

   Note: You can use this tab to import both product and categories changed records before going to update them.

Now you can use the previous procedure to update the imported records and have Optimizer build the static pages.

Updating Merchant Pages

Unfortunately, we don’t have a method as yet to automatically detect when a non-product or category page (Merchant Page) is changed. So, whenever you make changes to these pages, you should just run the Merchant Pages update.

To update Merchant Pages

1. In the Optimizer admin, click the Merchant Pages tab.
2. Click the Update button.

Additional .htaccess Rewrite Rules

Because of the disruptive nature of changing .htaccess rules, we strongly suggest you get assistance from an expert. Your best resource would be your store’s web host support personnel. (If they can’t help you, you need a new web host.)

There are a few items that an expert would find useful:

- When Optimizer runs and calls each page, it also includes a variable in its post called pm:optimizer which is equal to 1 (one). This lets you setup a rewrite rule depending on whether or not Merchant Optimizer is accessing the page or something else is accessing the page.
- If you have a single link to a Merchant page, product or category that you DO NOT WANT rewritten to point to the Optimizer created page, you can simply add: “NOOPT” to that url. For example,

  ```
  &mvt:global:sessionURL;Screen=PROD&Product_code=&mvt:global:product:
  code;&NOOPT
  ```

  This will cause Optimizer to leave the link as is. You might do this, for example, if the product needed to know that the customer was in a price group.

How Do I Know It Works

Because Optimizer pages look EXACTLY like Merchant pages, its hard to tell whether everything is working or not. Well, not exactly hard because Optimizer writes an HTML comment into every page it creates.

If you right click (in most browsers) on a web page, you should see a “View Page Source” command (or something similar). If you select that command and scroll to the bottom of the page, you should see something similar to:

```
!-- 06/28/2013: 20:02:05 EDT. URL:
http://www.phosphormedia.com/mm5/merchant.mvc?Screen=SFNT&pm:optimizer=
1 Generated by Merchant Optimizer. https://www.phosphormedia.com/ -->
```

This indicates that the page was created by Merchant Optimizer.

Getting the Most Benefit from Merchant Optimizer

The basic object of Optimizer is to improve search engine scanning by providing clean and fast loading static pages (with links that are search friendly) and to increase customer satisfaction (and better conversions) by increasing the responsiveness of your site while they are browsing.

These two tasks are not directly related but do overlap in deciding how to implement Optimizer pages.

There are two basic methods of using Optimizer:

- **As a Front End to Your Site**. This works great for SEO purposes, and provides some server performance benefit since over 40% of all page views are the first page users see.

- **For All Browsing**. This, works for great for SEO as well, but also gives you the maximum server performance benefit. By keeping customers in static product, category and common Merchant pages, you can reduce server load by 80% and increase conversion by letting customers browse in fast loading pages.

The following procedures assume that you have already created and reviewed your Optimizer pages in Staging Mode.

To use optimizer as a Front End

- Since this is the default method for Optimizer, just follow the procedure in “Switching to Live Mode”.

To use Optimizer for All Browsing

After using the procedure in “Switching to Live Mode”, you will need to add the following rewrite rules to your .htaccess file.

```
RewriteEngine On

RewriteCond %{QUERY_STRING} !pm:optimizer [NC]
RewriteCond %{QUERY_STRING} Screen=SFNT [NC]
RewriteRule ^(.*) http://www.domain.com/index.html? [R=301,L]

RewriteCond %{QUERY_STRING} !pm:optimizer [NC]
RewriteCond %{QUERY_STRING} Screen=PROD [NC]
RewriteCond %{QUERY_STRING} Product_Code=(.*)$ [NC]
RewriteRule ^(.*) http://www.domain.com/product/%1.html? [R=301,L]

RewriteCond %{QUERY_STRING} !pm:optimizer [NC]
RewriteCond %{QUERY_STRING} Screen=CTGY [NC]
RewriteCond %{QUERY_STRING} Category_Code=(.*)$ [NC]
RewriteRule ^(.*) http://www.domain.com/category/%1.html? [R=301,L]
```

Replace **www.domain.com** with your domain.

Replace **index.html** with the index file name you specified in the Optimizer Module Settings tab. (Generally there is no reason to have anything different.)

Replace the **/category/** directory with the Optimizer Category Page Directory you specified in the module’s Settings tab.

Replace the **/product/** directory with the Optimizer Product Page Directory you specified in the module’s Settings tab.

IMPORTANT: You need to enable .htaccess Mod Rewrite on the server. (Ask your host.)

Page Rendering Problems

Since Optimizer creates pages programmatically based on the HTML mark up on that page, some bad habits of HTML editing can cause problems. The following common errors, once corrected, will not only ensure better Optimizer pages, but, will also make your Merchant store more compatible with a larger number of browsers.

- When creating tables to contain images that must abut one another, keep that table, and all its cells on one line, and eliminate all extraneous spaces. For example:

  ```
  <table><tr><td><img src="start.gif”></td><td><img
  src=”middle.gif”></td><td><img src=”end.gif”></td></tr></table>
  ```

- When writing or copying Javascript code, make sure you use HTML comments to enclose the Javascript code. For example:

  ```
  <script language="JavaScript">
  <!--
  function PopWinSP(URL)
  {  popupWin = window.open(URL, 'Filename',
  'width=460,height=560,scrollbars=yes');
     window.top.name = 'opener';
  }
  //-->
  </script>
  ```

- Use correct quoting when creating Javscript Links. For example:

  ```
  <a href=’javascript:window.open(“foo.html”)’>
  ```

  is not correct. It should be

  ```
  <a href=”javascript:window.open(‘foo.html’)”>
  ```

- When writing or copying Javascript code, make sure all lines are correctly terminated with the JavaScript line termination character “;”. For example:

  ```
  function PopWinSP(URL)
  {  popupWin = window.open(URL, 'Filename',
  'width=460,height=560,scrollbars=yes');
     window.top.name = 'opener';
  }
  ```

Tips and Tricks

Here are some ideas to help you get more value from Merchant Optimizer.

Special thanks to David Hubbard of www.Hostasaurus.com and Remik of www.dotcomhost.com for their contributions to the htaccess rewrite rules.

Bypassing Dynamic Content on Static Pages

If you have customer specific dynamic content on your store front, category or product pages, you can easily bypass the display of this information when running Optimizer. For example, if you are using a “mini-basket” or a “Welcome Back: Customer Name” display, you can either replace them with something more generic such as a link to the basket or a general “Welcome to our store” message

The basic method is the same for all types of data: You add a special variable to the Optimizer page call settings, and then test for that variable before displaying the dynamic content. Here is an example of hiding a mini-basket display.

To bypass dynamic content

□ On the appropriate page template, you can wrap the dynamic content in an IF statement. For example:

```
<mvt:if expr="g.pm:optimizer">
   Items to appear on static pages only
<mvt:else>
   Items or code to appear on dynamic pages only
</mvt:if>
```

That’s it!

Making Search Engine Robots Bypass Merchant

While it may seem counterintuitive for us to suggest that you block search engines from scanning your site; that’s exactly what we do suggest. When search engines scan Merchant sites, they can use up a tremendous amount of server resources as they can “click” links (including add to basket links) several times a second.

...but what about search engine listings?

Your site will be more than adequately represented in search engines by the Optimizer produced pages, so in essence, there is no need at all for Search engines to crawl your Merchant generated pages. Also, having search engines index both Optimizer pages and the standard Merchant pages can cause your site to be penalized for “duplicate” content.

...so how do I create a robots.txt file?

Creating a Robots.txt to bypass Miva Merchant Pages

To create a robots.txt file to block search engines Miva Merchant pages, you’ll need a text editor and an FTP application to copy the file to your site.

To create the robots.txt file

1. Create a document in a text editor such as Note Pad, TextPad, or some other basic text editor.

IMPORTANT: Do not use an editor such as Microsoft Word or Word Pad.

2. Enter the following:

```
User-agent: *
Allow: /mm5/graphics/
Disallow: /mm5/
```

Where “mm5” is the sub-directory that your Merchant store runs from (what comes after the *yourdomain.com/* in your site’s normal URLs).

Shopability of Optimizer Pages

To make Optimizer pages friendlier to your shoppers, we suggest redirecting customers directly to the basket when they click an Add to Basket function. This is because the static nature of Optimizer pages cannot show an indication that the customer as added the product to their basket. In addition, you should add a Continue Shopping button to the Basket to allow customers to go back to the previous page.

Changing the Add to Basket Behavior

1. In the Merchant admin, open your Product Display template (Pages, Product Display).
2. Click Product Display Layout.
3. In the Product Display Layout template, scroll down and look for the following line: (Note, you may have to click Advanced Mode if the Product Display Layout template does not appear.)

   ```
   <input type=”Hidden” name=”screen” value=”BASK”>
   ```

4. Change that line to be:

   ```
   <input type=”Hidden” name=”screen” value=”PROD”>
   ```

5. Now, go to the Category Display template, and click the Category Product List Layout and do the same as the previous steps.
6. Important: If you make these changes BEFORE you create your Optimizer pages, you will have to re-run Optimizer before your pages will reflect this change.

Adding a Continue Shopping Button on the Basket

1. In the Merchant admin, open your Basket Contents page.
2. In the Basket Contents template locate the following code: (note, you might have to locate something similar as this feature changes from store to store)

   ```
   <div class="checkout-button"><input type="button" value="Checkout"
   onclick="window.location =
   '&mvte:global:secure_sessionurl;Screen=OINF&amp;Store_Code=&mvta:sto
   re:code;';" class="button" /></div>
   ```

3. Right before the preceding code, use the following HTML markup to create a “continue shopping” button. You may need to make additional changes to location and markup to reflect your store’s design:

   ```
   <div class="continue-button"><input type="button" value="Continue
   Shopping” onclick="javascript: history.go(-1)” class="button"
   /></div>
   ```

Optimizer Compatibility Issues

Understanding how Optimizer works is the key to understanding possible optimizer compatibility problems. Essentially, Optimizer looks at your store's screens in Miva Merchant the same as a browser would. It then copies the resulting page, replaces Merchant specific links, removes the other dynamic specific values (such as the session id), and then writes that file to your server.

3rd Party Modules

Since Optimizer its calling dynamic pages and creating standard (or static) HTML pages, features such as Related Products from Order History, while captured at the moment Optimizer runs, do not update with subsequent orders until you re-run Optimizer. This would also include modules such as "review this product" type modules and discounts based on customers. However, most modules that link back to Merchant to access their features work just fine. For example, Tell A Friend, Wish Lists, discounts by product or category, etc.

Javascript Issues

In addition to making sure any Javascript or other scripting code is properly formatting with line termination characters, it's a good idea to use an "include" method for javascripts. For example, instead of adding the javascript code to the global header as:

```
function PopWinDN(URL) {
      popupWin = window.open(URL, 'Filename', 'scrollbars,resize="yes",
width=540,height=600')
      window.top.name = 'opener';
}

function PopWinSP(URL) {
      popupWin = window.open(URL, 'Filename',
'width=460,height=560,scrollbars=yes')
      window.top.name = 'opener';
}
```

You could place the script for the code in a file called popup.js, and reference that file as:

```
<SCRIPT LANGUAGE=JAVASCRIPT SRC=/popup.js></SCRIPT>
```

Need Help? (aka Getting Support)

Here at Phosphor Media, we are fanatical about providing support; and our customers agree. We have a reputation of providing some of the best support in the Miva Merchant community. However, support is a relationship and like all successful relationships, both sides need to put forth an effort and strive to communicate effectively.

To this end, every product we sell has a built-in support ticket feature. Using this support feature is by far the fastest, most effective way for us to help you. The other key to getting fast solutions is to invest a bit of time when describing your issue. If you run into a problem please include the steps you took before the problem appeared. If it's a general question that is NOT answered by our documentation, tell us what you are trying to accomplish - not just what you are doing.

If You Can't Get to the Internal Support Feature

If for some reason you cannot use the internal support feature, the next best method is to use the support form on our web site:
http://www.phosphormedia.com/support/