

Mambo-phpShop 1.2

Tutorial

Author: Russell Walker (www.netshinesoftware.com)

Version: 1.0

Last Updated: 27/01/2005

Copyright Notice:

© Copyright 2005 Netshine Software Limited. All rights reserved.

You may print this document only for your own use. You may not distribute this document in any form – printed or electronic – without the express permission of the publisher. You may not link directly to this document without the express permission of the publisher. You may however, provide a link to any page in the publisher's website (www.netshinesoftware.com) to allow the document to be downloaded from there. This document is provided free of charge, and without any warranty of any kind. Netshine Software Limited accepts no liability for any loss or damages resulting from the use of this document or any of the information or opinions it contains.

Contents

Introduction.....	3
Basic Principles – Important!.....	3
Getting Started.....	3
Figure 1 – Accessing the Site Module Manager.....	4
Figure 2 – The Site Module Manager.....	4
Figure 3 – The Site Module Editor.....	5
Figure 4 – Example of Module Output in Website Front End.....	6
Figure 5 – The Parameters Tab.....	6
Configuration.....	6
Figure 6 – Accessing mambo-phpShop.....	7
Figure 7 – Store Summary.....	7
Figure 8 – Accessing the Store Editor.....	8
Figure 9 – The Store Editor.....	9
Figure 10 – Configuration.....	10
Figure 11 – Accessing The VAT Rate List.....	11
Figure 12 – The Tax Rate List.....	11
Manufacturers and Vendors.....	12
Figure 13 – Accessing the Manufacturers List.....	12
Figure 14 – The Manufacturer List.....	13
Figure 15 – Manufacturer Information.....	13
Figure 16 – Accessing the Vendor List.....	14
Shipping Details.....	14
Figure 17 – The Shipper Editor.....	15
Figure 18 – The Shipping Rate Editor.....	15
Setting Up Your Catalog.....	16
Figure 19 – Accessing The Category Tree.....	16
Figure 20 – The Category Tree.....	17
Figure 21 – The Category Editor.....	18
Figure 22 – Accessing the Product List.....	19
Figure 23 – The Product List.....	19
Figure 24 – The Product Editor, Product Information Tab.....	20
Figure 25 – Adding a New Attribute.....	21
Figure 26 – The Attribute List.....	21
Figure 27 – The Product Editor, Product Status Tab.....	22
Figure 28 – Example of Attributes in the Front End.....	23
Figure 29 – The Product Editor, Dimensions and Weight Tab.....	23
Figure 30 – The Product Editor, Product Images Tab.....	24
Figure 31 – Adding a New Item.....	25
Figure 32 – Product List with Items.....	25
Figure 33 – Return to Parent Product Toolbar Button.....	26
Payment Methods.....	26
Figure 34 – The Payment Methods List.....	26
Figure 35 – The Payment Method Editor.....	27
Order Processing.....	27
Figure 36 – The Order Status List.....	28
Figure 37 – The Order Details.....	29

Introduction

The following is a brief introduction to the mambo-phpShop component which provides direct integration between the Mambo Content Management System and the phpShop open source e-commerce software – enabling you to manage both website content and an e-commerce catalogue through a single familiar interface. Text presented in bold is just to highlight key words or points, and to help you scan quickly through the document.

This guide assumes that you already have Mambo installed on your web server. For information about how to install Mambo, see the official documentation and the forums on the Mambo website (see www.mamboserver.com). For an introduction to using Mambo, it is highly recommended that you read the Mambo Quick Start Guide before reading this tutorial. The Mambo Quick Start Guide can be found here: www.netshinesoftware.com/mambo-tutorial.html.

Installation of mambo-phpShop is also beyond the scope of this guide, although the process is quite simple if you are able to use the automatic installer. For details on how to install mambo-phpShop, please refer to the official website: www.mambo-phpshop.net and read the user guide (available by selecting 'documentation' on the main menu of the official website).

Basic Principles – Important!

The mambo-phpShop component does not in itself provide e-commerce facilities – the underlying system is a separate e-commerce package: **phpShop**. The mambo-phpShop component provides a way of administering the phpShop package via the mambo administration tool. This is important to understand, because it means that all of the data associated with your online shop is held separately from the data associated with your mambo website.

The sections, categories, and items that make up your mambo website therefore have absolutely nothing to do with the category structure you use for your online shop. One of the main drawbacks of Mambo 4.5.1a (and previous versions) is that you are limited to a 3-level hierarchy (section->category->item) – you cannot have **nested categories** (eg. section->1st-category->2nd-category->3rd-category->item). With mambo-phpShop however, you can have as many nested categories as you like – the hierarchy is completely up to you.*

Although much of the mambo-phpShop component is quite self-explanatory, there are so many options and features that it can be difficult to know where to start, or in what order you should do things. If you follow this guide from beginning to end, you should manage to not only become familiar with the way the product works, but also to build your shop in a logical and practical order.

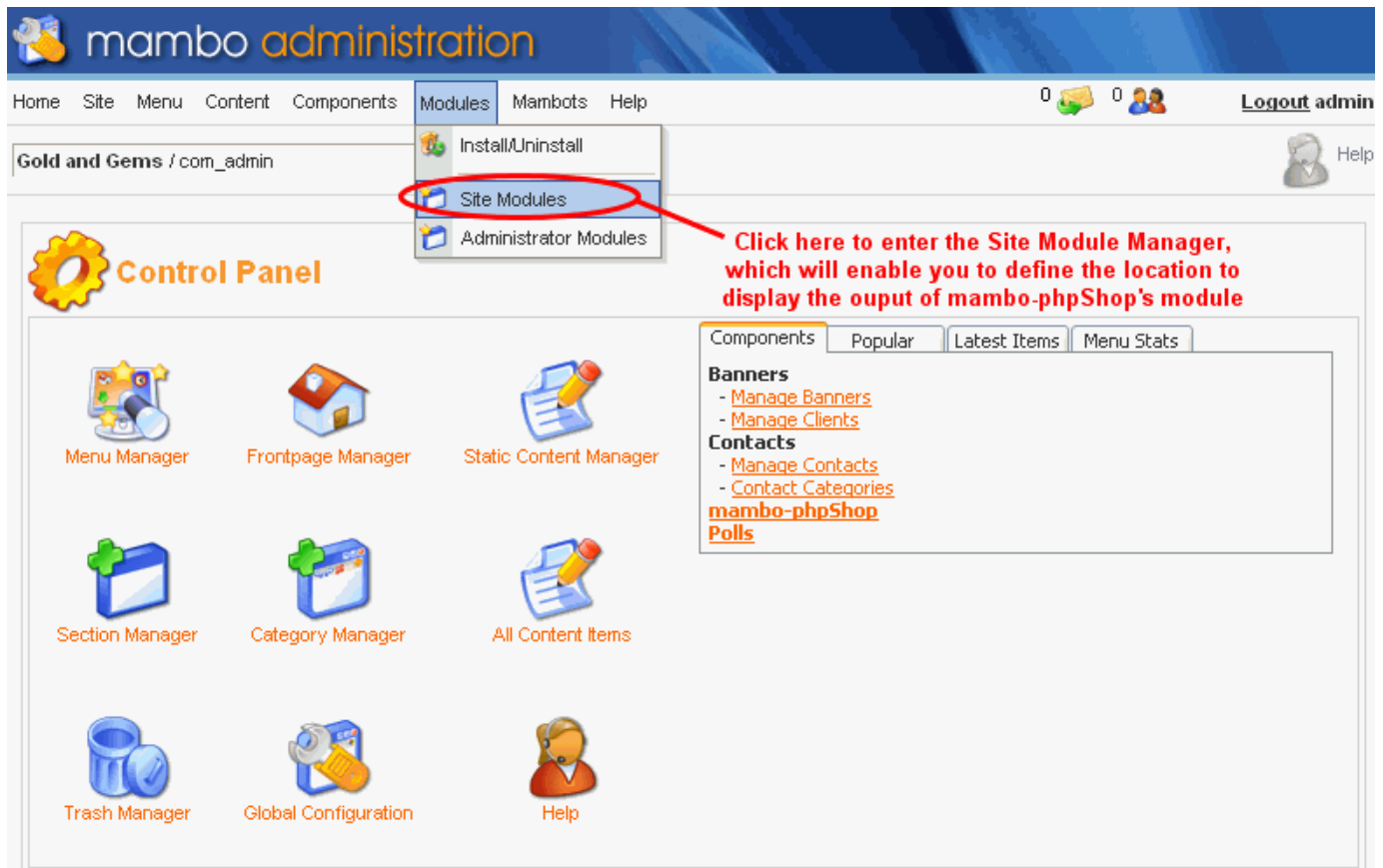
Getting Started

Installation of mambo-phpShop consists of 2 stages: **component installation**, and **module installation**. In mambo, **modules** are individual packages of 'screen real estate' and are usually used for displaying the output of a **component**. The component is what does the bulk of the work, but the module is what enables you to see the results. It is therefore essential to make sure you install the module as well as the component!

As with applying templates (see the Mambo Quick Start Guide), simply installing mambo-phpShop does not make it appear on your website – it just makes it available to you. Once installed, you need to indicate to mambo that you want to display the output of the module on your site, as well as *where* and *how* to display it. To do this, go to Modules->Site Modules as shown in figure 1.

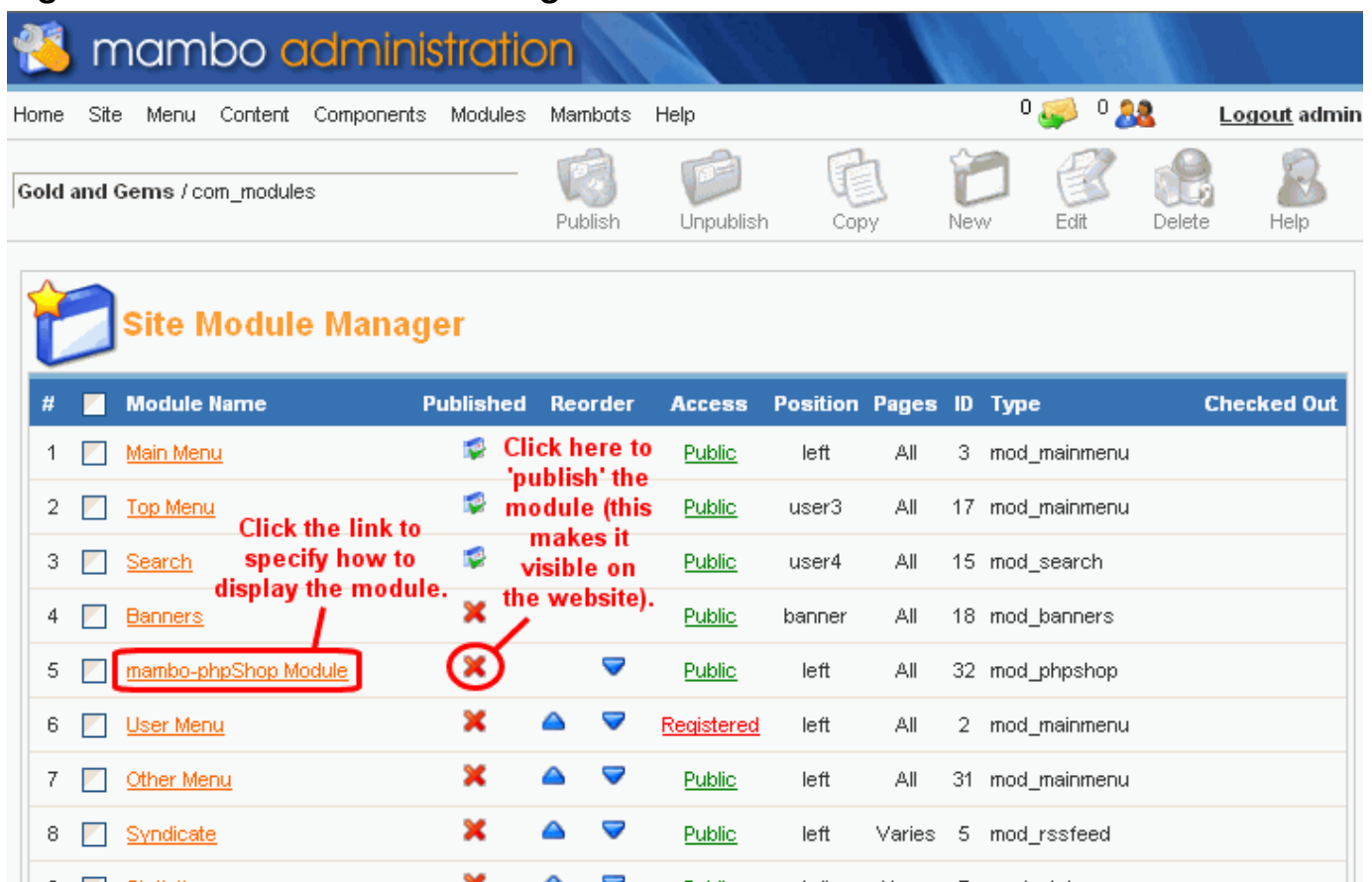
* It is expected that a future version of Mambo will allow multiple nested categories, but until then, many have found that mambo-phpShop allows them to work around this limitation – even if they do not actually want to sell anything online. The shopping cart features of mambo-phpShop can be turned off, thus becoming simply a hierarchical catalogue with no shopping cart or payment facilities.

Figure 1 – Accessing the Site Module Manager



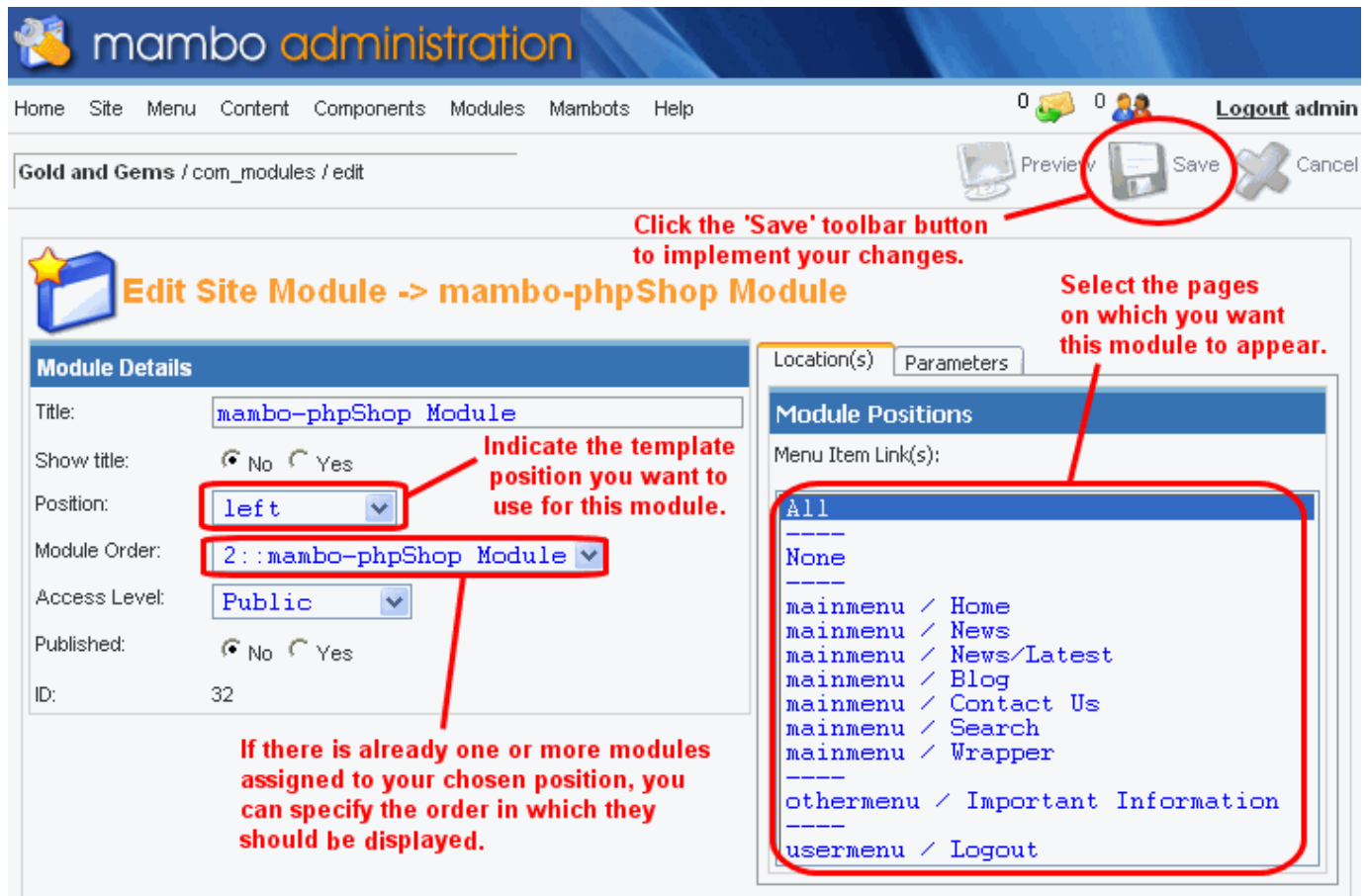
This takes you to the **Site Module Manager** as displayed in figure 2.

Figure 2 – The Site Module Manager



In order for the output from a module to be displayed at all in the front end of your website, it has to be 'published'. You can publish the mambo-phpShop module by clicking on the red 'X' next to the module name in the Site Module Manager, or alternatively, check the box next to the module, and then click on the 'Publish' toolbar button. Click on the module name to enter the **Site Module Editor** (see figure 3).

Figure 3 – The Site Module Editor



Here you can optionally enter a **title for the module** and whether that title should be displayed to the end user, as well as specifying the **template position** where you would like the output to be displayed.

It is important to note that most templates do not support all of the available positions, so some of the options available in this dropdown list will actually result in the module not appearing at all. The most common template positions (left, right, top, bottom) are almost always supported, although it has been known for a template to insist that the 'left' position actually be located on the right of the page!

When using this editor, you need to be aware that what you are dealing with is specifying the location of your **product catalogue hierarchy display**. Basically, this forms a menu to enable your visitors to navigate to the products in your shop (see figure 4). It is therefore usually appropriate to locate this module in the same place (or near to) your main menu (which is also a module). The actual **product details** will be displayed in the main area of your web pages – regardless of the location you specify for the module (but without selecting something in the menu provided by this module, the user will never be able to get to the product details).

The 'Location(s)' tab on the right of the Site Module Editor lists all of the menu options from all of your menus. This enables you to **specify which pages of your site will display this module**. For example, you might want your shop hierarchy only to be visible from your home page. To specify more than one page, hold down the 'Ctrl' key whilst selecting (or just use the 'All' option if you want the module output to be visible on every page – which is quite likely for a shop).

Click on the 'Parameters' tab for more options regarding the output of this module (see figure 5).

Figure 4 – Example of Module Output in Website Front End

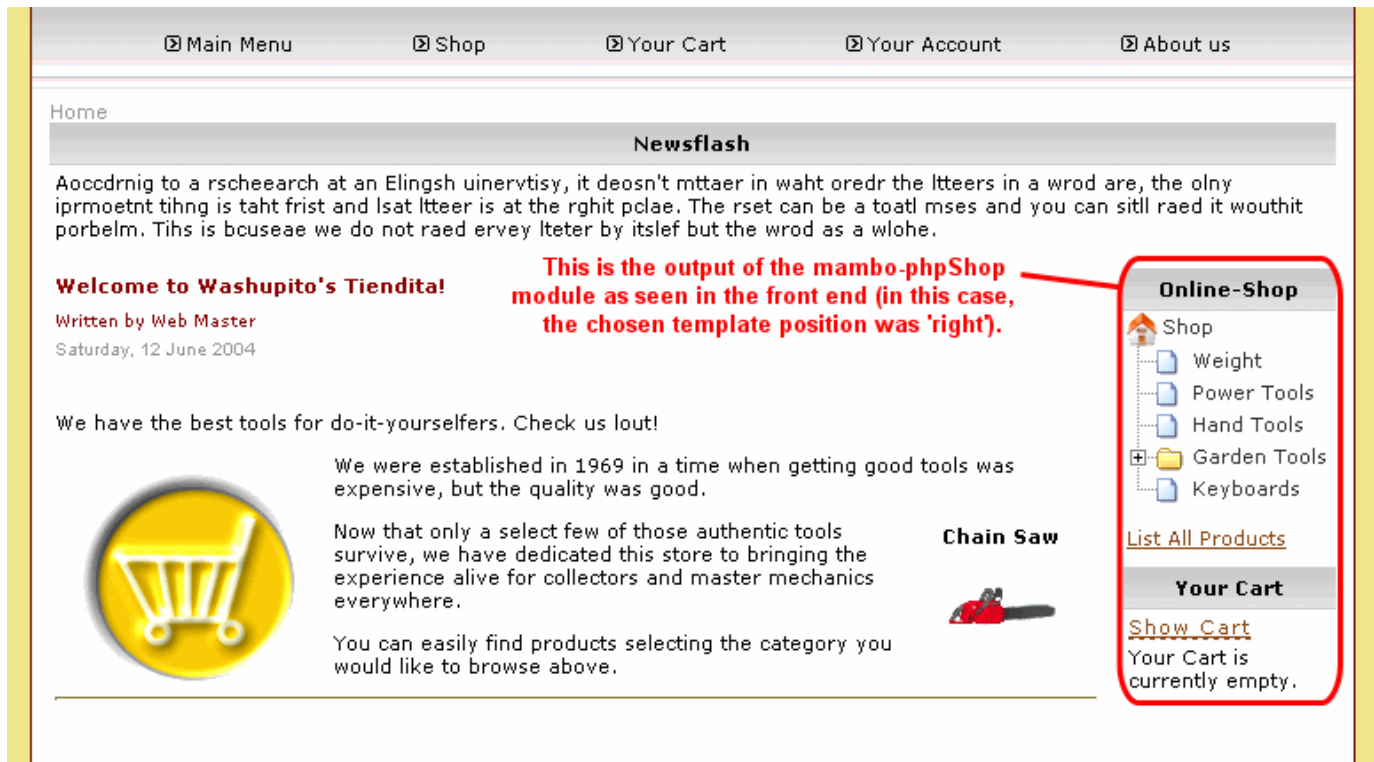
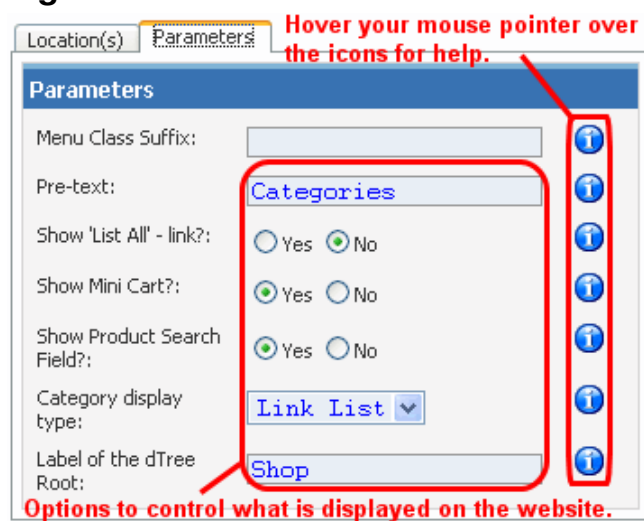


Figure 5 – The Parameters Tab



Most of these options are fairly self-explanatory, but it is worth highlighting the '**Category display type**'. The '**Link List**' option will display the hierarchy using menu links in the same style as your template, whereas '**dTree**' will present your hierarchy using a graphical tree structure (as seen in figure 4). If your template has a dark background, you will probably find the 'Link List' style more appropriate, as the images of the graphical tree are 'antialiased' against a light background (so the pictures look strange against a dark background).

Configuration

You can enter the **administration area** for your online shop, by selecting 'mambo-phpShop' from the components menu, or from the link on the home page of mambo, as shown in figure 6. This takes you to the **store summary** screen, as shown in figure 7. Note, you can return to the store summary screen any time by selecting 'summary' from the 'Store' menu in the mambo-phpShop menu bar.

Figure 6 – Accessing mambo-phpShop.

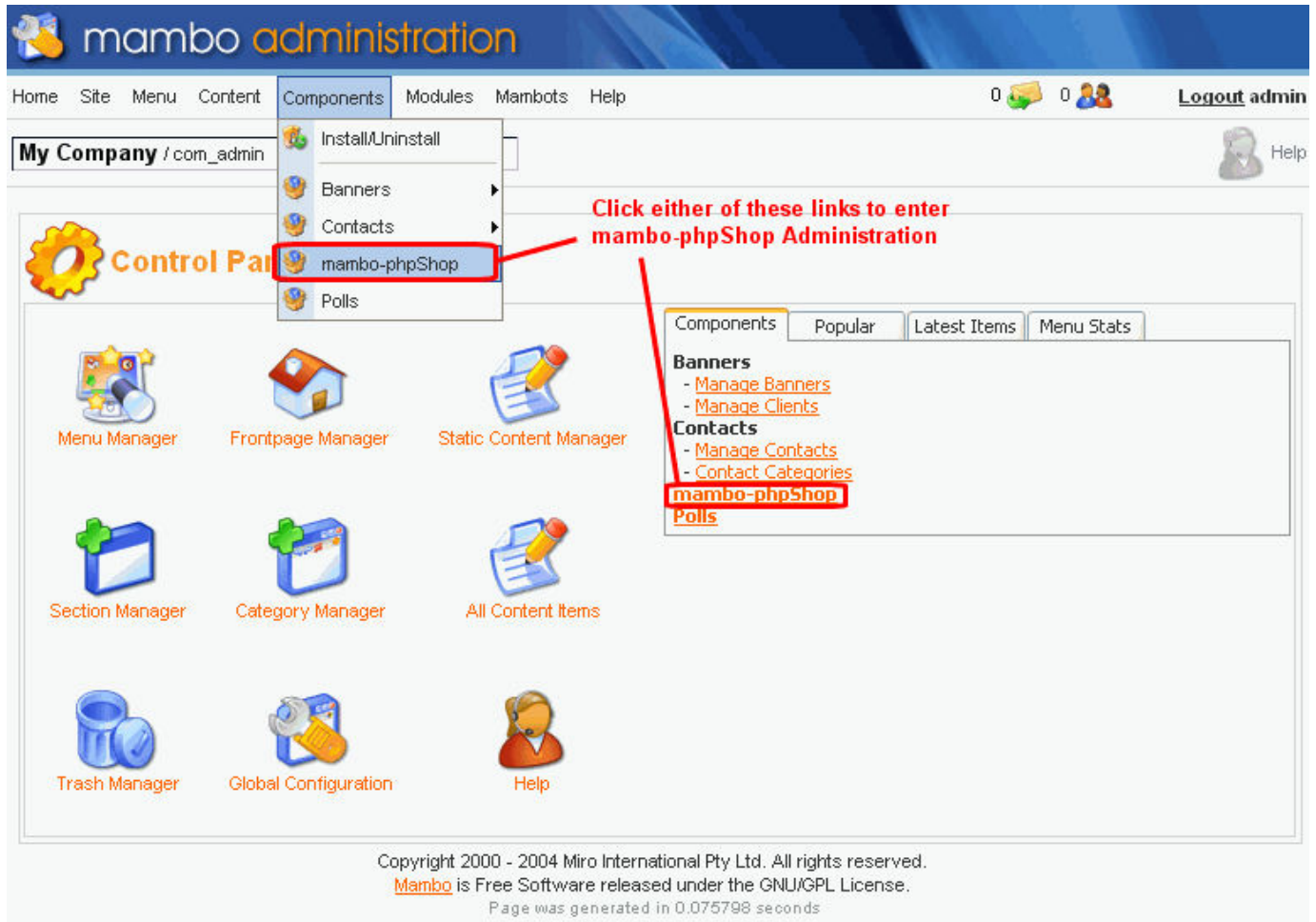
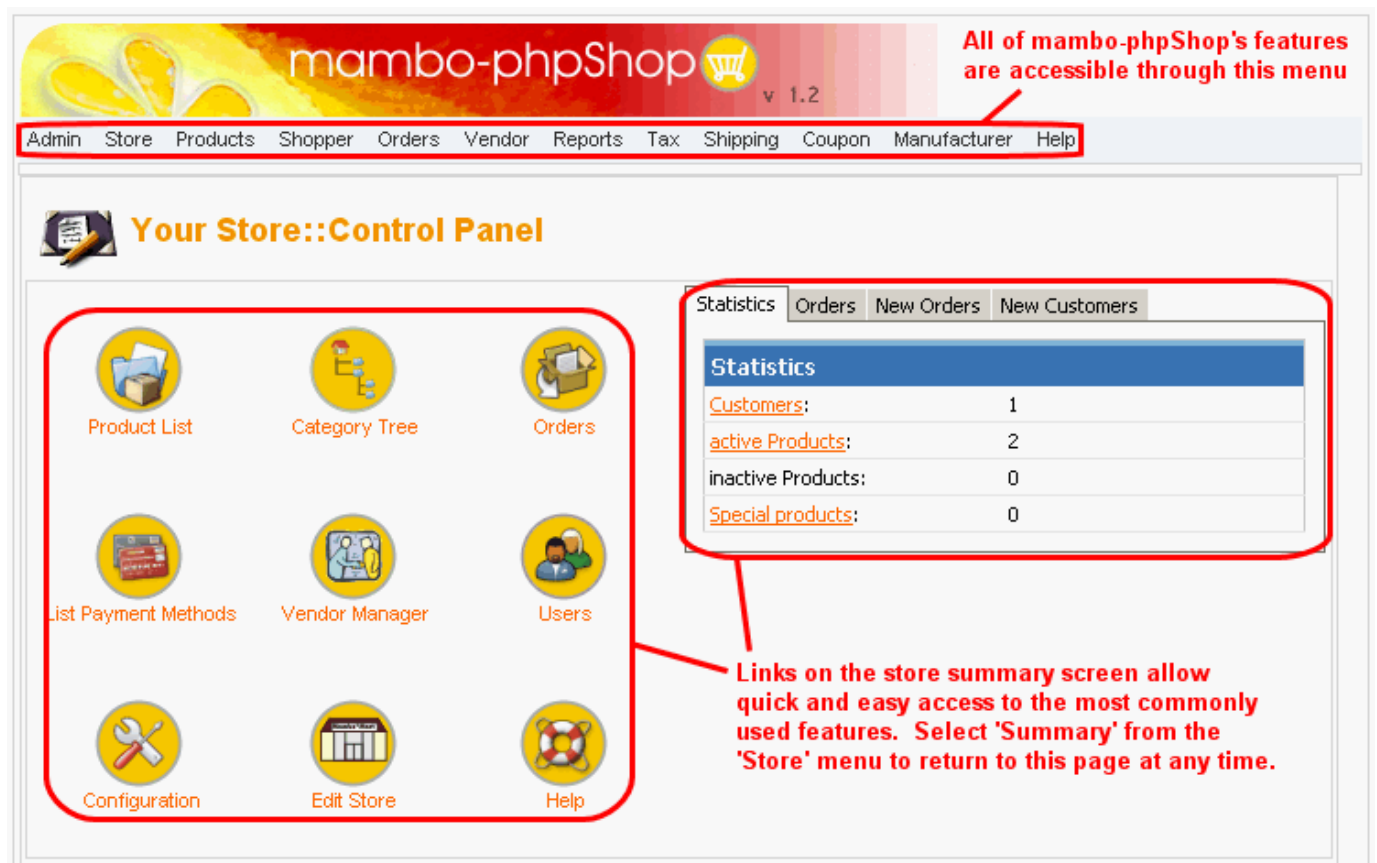


Figure 7 – Store Summary



Good news! From here on, the majority of features are well documented in the program itself, so administering your store is very intuitive – a bit of experimentation, and you should be well on your way. One of the first tasks is to set

up your basic **store configuration** – things like store name, and default currency. Select 'Edit Store' from the 'Store' menu to set up these details (see figure 8).

Figure 8 – Accessing the Store Editor

The screenshot shows the Admin Control Panel for mambo-phpShop v 1.2. The 'Store' menu is open, with 'Edit Store' highlighted. A red box highlights the 'Edit Store' menu item, and a red circle highlights the 'Edit Store' icon in the main dashboard. A red arrow points from the menu item to the icon. A text box with a red border contains the instruction: 'Click on the menu option or the icon to enter the Store Editor.'

Statistics

Statistics	Orders	New Orders	New Customers
Customers:	1		
active Products:	2		
inactive Products:	0		
Special products:	0		

Fill in as many of the details as you can on each tab in the dialog box shown in figure 9.


Figure 9 – The Store Editor

mambo-phpShop v 1.2

Admin Store Products Shopper Orders Vendor Reports Tax Shipping Coupon Manufacturer Help

Store Information

Store **Store Information** Contact Information

Full Image: 

Upload Image:

Currency:

Minimum purchase order value for your store:

Minimum Amount for Free Shipping: ⓘ

Currency Display Style

Currency symbol : ⓘ

Decimals : ⓘ

Decimal symbol : ⓘ

Thousands separator : ⓘ

Positive format : ⓘ

Negative format : ⓘ

Description: Enter a description of your online shop. This is displayed as the shop's home page.

Terms of Service: Enter your legal mumbo-jumbo here!

Use the 'Save' toolbar button (not shown) to confirm your changes.

Upload your logo - this is displayed in various places, including order summaries. The image will only be displayed here after you save these details and come back into the store editor.

Hover your mouse pointer over the question mark icons for more information about each option.

On some of the dialogs in the mambo-phpShop administration tool, after you click on the 'Save' toolbar button, you are taken to **a read-only summary** of the details you have just saved, along with a link to go back and edit the item again. Remember to check all of the tabs on each dialog you enter, to make sure you have filled out everything.

After setting up the basic details about your store in the store editor, you need to go to the **configuration** area. To do this, select 'Configuration' from the 'Admin' menu. There are various options here, allowing you to specify which features you want to use, and how you want to display things. See figure 10.

Figure 10 – Configuration

The screenshot shows the 'Configuration' page in mambo-phpShop v 1.2. The top navigation bar includes 'Admin', 'Store', 'Products', 'Shopper', 'Orders', 'Vendor', 'Reports', 'Tax', 'Shipping', 'Coupon', 'Manufacturer', and 'Help'. The main content area is titled 'Configuration' and features a status message: 'Configuration file status: phpshop.cfg.php is writeable'. Below this is a tabbed interface with tabs for 'Global', 'Path & URL', 'Site', 'Shipping', 'Checkout', and 'Downloads'. The 'Global' tab is active, showing two sub-tabs: 'Global I' and 'Global II'. The 'Global I' sub-tab contains several configuration options:

- Use only as catalogue**:
- Show Prices**:
- Show Prices including tax?**:
- Virtual Tax**:
- Tax mode:**
- Enable multiple tax rates?**:
- Subtract payment discount before tax/shipping?**:

At the bottom, there is an 'ENCODE KEY' field with a blue information icon and the value 'mambophpShopIsCool'. A large red callout box on the right side of the configuration options contains the following text:

If you check this, you disable all cart functionalities.

Check to show prices. If using catalogue functionality, some don't want prices to appear on pages.

Sets the flag whether the shoppers sees prices including tax or excluding tax.

This determines whether items with zero weight are taxed or not. Modify ps_checkout.php->calc_order_taxable() to customize this.

This determines which tax rate is taken for calculating taxes:

- the one from the state / country the store owner comes from
- or the one from where the shopper comes from.

Check this, if you have products with different tax rates (e.g. 7% for books and food, 16% for other stuff)

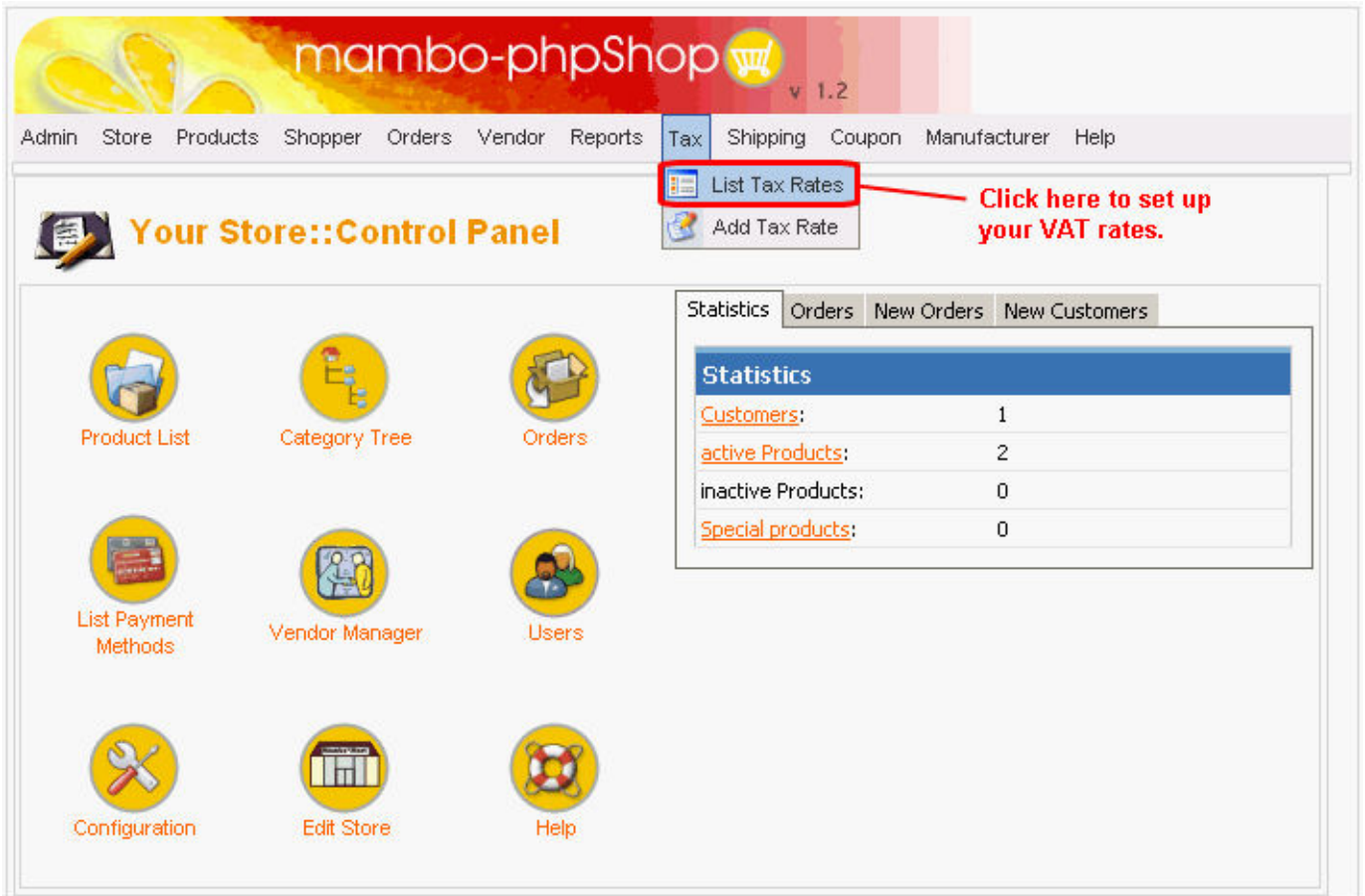
Sets the flag whether to subtract the Discount for the selected payment BEFORE (checked) or AFTER tax and shipping.

Red annotations on the screenshot include:

- A red box around the 'Global' tab with the text: 'Make sure you check all of the tabs on every page.'
- A red box around the 'Global I' sub-tab with the same text: 'Make sure you check all of the tabs on every page.'
- A red circle around the information icon with the text: 'All of the options have explanatory text to help you. If in doubt, experiment!'

After setting up your configuration details, you need to specify how you want to charge **VAT** (if at all). To do this, you need to access the list of tax rates, by selecting '**List Tax Rates**' from the 'Tax' menu, as shown in figure 11.

Figure 11 – Accessing The VAT Rate List



Unless you are trading to and from various different locations in the world, you will probably only want a single tax rate – or maybe two (eg. one for within the European Union, and one for the rest of the world). You can have as many different rates as you like though – so could set a different rate for each country if you want. The list of rates for your shop is displayed as shown in figure 12.

Figure 12 – The Tax Rate List



Click on one of the links, or the 'New' toolbar button to add or edit a tax rate. You will need to specify the country and region, as well as the percentage tax applicable. The tax percentage must be entered as a decimal value (so the current UK rate of 17.5% VAT is represented as 0.175).

Manufacturers and Vendors

If your shop only includes products from one manufacturer AND there is only one individual or company selling products through your website, then you can skip this section and move on to 'Setting Up Your Catalogue'. Even if you have many different manufacturers, you might not want to list details about them on your site, in which case you too can skip this section.

For those still with me, one of the interesting features of mambo-phpShop is the ability to categorise manufacturers, and even to have a number of little 'sub-shops' hanging off your website. This latter feature might be useful if your site represents a chain of stores, a franchise, or a membership system where members can sell their own products.

Just to clarify: a **manufacturer** is an individual or company that produces the merchandise your site sells. A **vendor** is an individual or company that sells goods through your website.

In the majority of cases, there will be several manufacturers, but only one vendor. If there is only one manufacturer and only one vendor in your website, you need to nothing – a single manufacturer record and a single vendor record are supplied by default. If you have several vendors, but each one manufactures their own products, you will likely need to add one record to your manufacturers list AND one record to your vendors list for each vendor.

If you wish, you can choose to categorise your manufacturers. You can have as many **manufacturer categories** as you like, but manufacturer categories (unlike product categories, which we will come onto later) cannot be nested – in other words, there is no hierarchy. You just have a flat list of categories, and each category has a flat list of manufacturers. Use the 'List Manufacturer Categories' option from the 'Manufacturer' menu to define your manufacturer categories if required.

It is also possible to create manufacturer records directly without specifying a category. To create your manufacturers, select 'List Manufacturers' from the 'Manufacturer' menu, as illustrated in figure 13.

Figure 13 – Accessing the Manufacturers List




The screenshot shows the mambo-phpShop v 1.2 control panel. The 'Manufacturer' menu is open, and 'List Manufacturers' is highlighted with a red box. A red arrow points from this menu item to a red text box that says 'Click here to access the list of Manufacturers.' The control panel also features a 'Your Store::Control Panel' section with various icons for Product List, Category Tree, Orders, List Payment Methods, Vendor Manager, Users, Configuration, Edit Store, and Help. A 'Statistics' table is visible in the background.

Statistics	
Customers:	1
active Products:	2
inactive Products:	0
Special products:	0



This takes you to the **manufacturer list**, shown in figure 14.

Figure 14 – The Manufacturer List

My Company / [com.phpshop](#) / cancel Click here to create a new manufacturer. 

Manufacturer List Search: Click this icon to delete a manufacturer.

Click either of these links to edit an existing Manufacturer.

#	Manufacturer Name	Admin	Remove
1	Gems Unlimited	Update	
2	Goldsmith UK	Update	

Click on the 'New' toolbar button to create a new manufacturer, or on one of the links displayed to edit an existing manufacturer record. You can enter the manufacturer's name, website address, category (if applicable), e-mail address, and description using the dialog shown in figure 15.

Figure 15 – Manufacturer Information

Add Information

Manufacturer Information

Manufacturer Name:

URL:

Manufacturer Category:

Email:

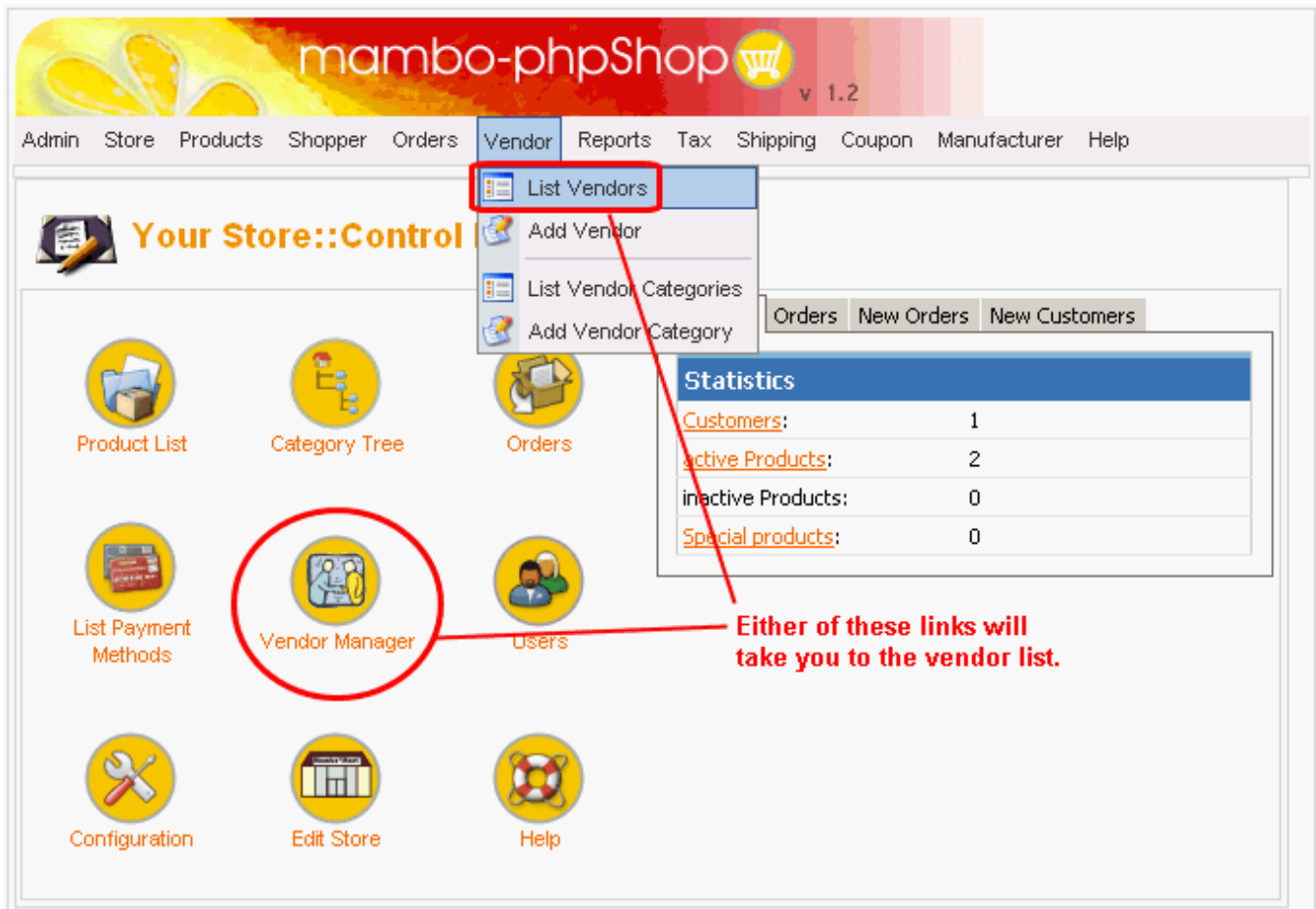
Description:

Click on the 'Save' toolbar button (not shown) to confirm your changes.

Setting up vendors is similar, but you can add a bit more information than with manufacturers. In fact, the vendor information screens are practically identical to the store editor screens (see figure 9). As with manufacturers, you can set up **vendor categories** by selecting 'List Vendor Categories' from the 'Vendor' menu.

To set up the actual vendors, select 'List Vendors' from the 'Vendor' menu, or use the icon on the summary page as shown in figure 16.

Figure 16 – Accessing the Vendor List



As usual, this takes you to a list of existing records (if any), from where you can add or edit your vendors. Note: In the version of mambo-phpShop in use at the time of preparing this tutorial, there appears to be a bug which prevents you from entering a detailed description and terms of service for your individual vendors.

Shipping Details

I know you're anxious to start adding products to your catalogue, but please be patient! It is wise first of all to think about how you will charge for **postage and packing**. The easiest way is to have a fixed shipping price for all of your products, but this is not always practical – especially if your products vary greatly in size and/or weight.

Not only do you need to think about shipping rates, but also shipping methods. Will you offer more than one shipping method – eg. next day delivery as well as standard? DHL as well as Royal Mail?

Even if you decide to only use one company for all of your shipping, you still need to tell mambo-phpShop about it. To set up a new 'shipper', select 'Create Shipper' from the 'Shipping' menu. This takes you to shipper editor, shown in figure 17.

Figure 17 – The Shipper Editor

List order is just a numeric value which is used to sort the shipper names, so enter 1 for the shipper that you want to be listed 1st, 2 for the one you want to be listed 2nd, etc.

When you have added the shipping companies you want to use, you need to configure the **shipping rates**. You can set up different rates for different service levels (eg. next day, two day, first class etc.) or according to the weight of the shipment. You can also select whether to add VAT to the shipping rate or not. All of this takes place in the Shipping Rate Editor – see figure 18.

Figure 18 – The Shipping Rate Editor

To edit or delete existing shipping rates, select 'Shipping Rates' from the 'Shipping' menu.

Setting Up Your Catalogue

Now we get to the heart of your online store – the products. Before you can actually add the products though, you need to decide on a **category structure** that is logical, and will assist your users to locate the product they are looking for. To set up a category tree, select 'List Categories' from the 'Products' menu, or use the summary page icon as shown in figure 19.


Figure 19 – Accessing The Category Tree



This will take you to the Category Tree, as illustrated in figure 20.

Figure 20 – The Category Tree

Home Site Menu Content Components Modules Mambots Help 0 0 [Logout admin](#)

My Company / com_phpshop Click the 'New' toolbar button to create a new category. 












mambo-phpShop v 1.2

Admin Store Products Shopper Orders Vendor Reports Tax Shipping Coupon Manufacturer Help

Category Tree

You can access the products in each category directly.

Use these buttons to specify the order in which to display the categories.

#	Category Name	Category Description	Products	Publish?	List Order	Remove
1	__1 Rings		0 Products [Show]	✓	▼	
2	__2 Diamond		2 Products [Show]	✓	▼	
3	__3 Wedding Bands		0 Products [Show]	✓	▼	
4	__3 Engagement Rings		0 Products [Show]	✓	▲ ▼	
5	__4 Certified		0 Products [Show]	✓		
6	__3 Three-stone Rings		0 Products [Show]	✓	▲	
7	__2 Ruby		0 Products [Show]	✓	▲ ▼	
8	__2 Emerald		0 Products [Show]	✓	▲	
9	__1 Necklaces		0 Products [Show]	✓	▲ ▼	
10	__1 Pendants		0 Products [Show]	✓	▲	
11	__2 Gems		0 Products [Show]	✓		

Tree structure represented by indentation and numeric identification of hierarchy level.

Click a link to edit the required category.

Use this icon to delete a category.

Again, the layout of the screen is fairly self-explanatory. Clicking on a category name allows you to edit the details for that category, a separate link takes you to a list of products that are assigned to the category, and you can adjust the order in which sibling categories are displayed.

When you click on the 'New' toolbar button, or on a category name, you are taken to the **Category Editor**, which is illustrated in figure 21.

Figure 21 – The Category Editor

The screenshot shows the 'mambo-phpShop v 1.2' interface. At the top right, a red arrow points to a 'Save' button with the text: 'Use the 'Save' toolbar button (not shown) to confirm your changes.' The navigation menu includes: Admin, Store, Products, Shopper, Orders, Vendor, Reports, Tax, Shipping, Coupon, Manufacturer, Help.

Category Information

Publish?: *The category will only appear on your website if this box is ticked.*

Category Name: Diamond

Category Description: *The description you enter here forms the 'home page' for this category.*

These are the names of template files that will be used to display this category's details. Normally, you will want to leave them as their default values.

ListOrder: 1. Diamond

Parent: -|- Rings *This is where you specify the 'parent' category for this category - thus constructing your hierarchical tree of categories.*

Category Browse Page: browse_1

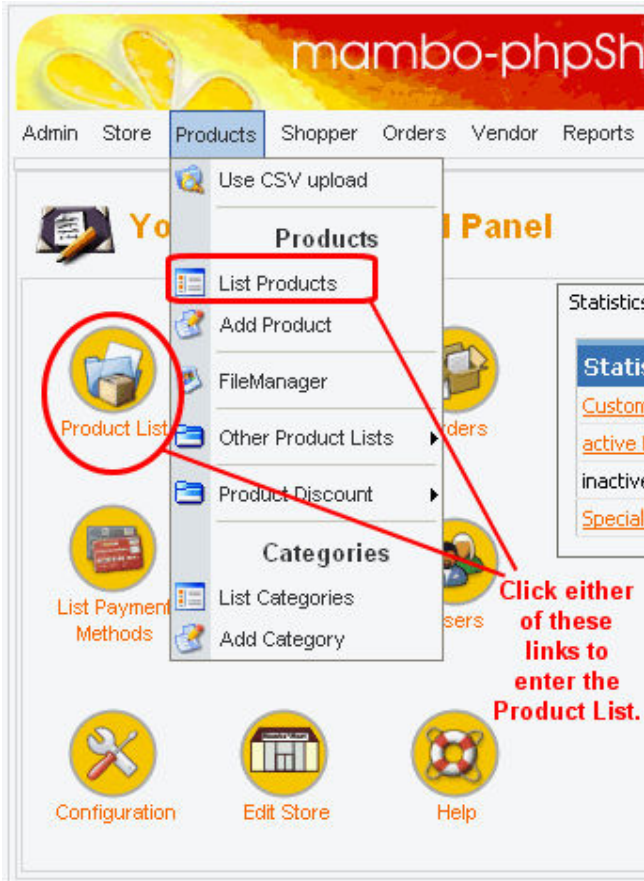
Show x products per row: 1 *Specify the number of products to display per row within this category.*

Category Flypage (leave BLANK if you have no individual php-file for it!): shop.flypage

In the vast majority of cases, when you create a new category, you will only need to enter a name and description for the category, and specify the **parent category**. Don't worry if you forget to specify the parent – you can just click on the category name to come back into the above screen and edit it.

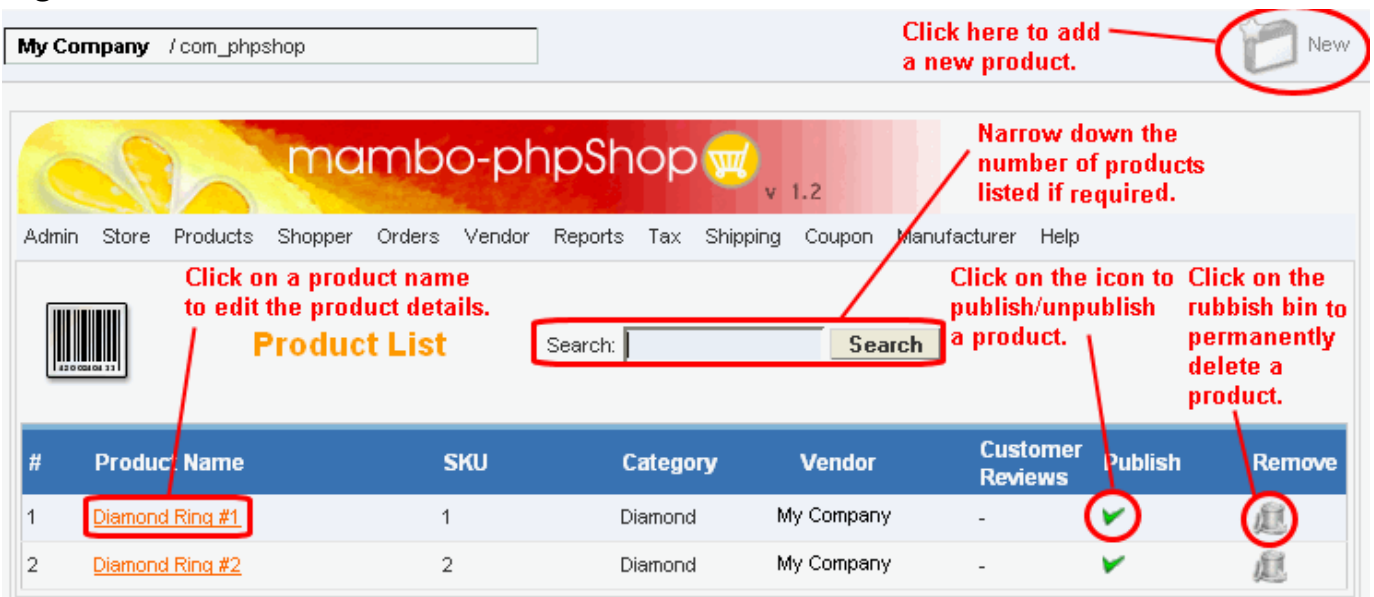
After you have set up some categories, it is time to add the **product details**. To do this, select 'List Products' from the 'Products' menu, or click on the **'Product List'** icon on the store summary, as shown in figure 22.

Figure 22 – Accessing the Product List



This will take you to a list of all products in the entire database. If you wanted to narrow this down to only those products in a particular category, you could click on the '[Show]' link in the category tree (figure 20). Either way, you can use the **search box** to filter down to the products you are interested in. Of course, this is not a concern until your catalogue is fairly well populated. Figure 23 shows the product list, which is in a format that you should find familiar by now!

Figure 23 – The Product List



When you click on a product name, or create a new product, you are taken to the **Product Editor**. This is made up of several tabs (the first of which is shown in figure 24).

Figure 24 – The Product Editor, Product Information Tab

The screenshot shows the 'Update Product' interface in mambo-phpShop v 1.2. The top navigation bar includes Admin, Store, Products, Shopper, Orders, Vendor, Reports, Tax, Shipping, Coupon, Manufacturer, and Help. The main content area is titled 'Update Product' and features a 'Preview product flypage in shop' button. Below this are tabs for Product Information, Product Status, Product Dimensions and Weight, and Product Images. The Product Information tab is active, showing fields for Publish?, SKU, Name, URL, Category, Category 2, Vendor, Manufacturer, Retail Price, VAT Id, and Short Description. A 'Flypage Description' field is also present. Red annotations provide detailed explanations for several fields: 'Publish?' (checked), 'SKU' (1), 'Name' (Diamond Ring #1), 'URL' (empty), 'Category' (Diamond), 'Category 2' (Select), 'Vendor' (My Company), 'Manufacturer' (test), 'Retail Price' (4581.23 British Pound), and 'Short Description' (A very expensive ring!). The 'Flypage Description' field contains a paragraph of text. A rich text editor toolbar is visible at the bottom right.

Update Product

Admin Store Products Shopper Orders Vendor Reports Tax Shipping Coupon Manufacturer Help

Product Information Product Status Product Dimensions and Weight Product Images

Publish?: **SKU:** 1 **Name:** Diamond Ring #1 **URL:** **Category:** Diamond **Category 2:** Select **Vendor:** My Company **Manufacturer:** test **Retail Price:** 4581.23 British Pound **VAT Id:** 3 (0.1750%) **Short Description:** A very expensive ring!

Flypage Description: This beautiful masterpiece is worth every penny of the exorbitant price tag. Believe me, if you buy this, I will be very happy.

Annotations:

- Click here to preview what the product details page will look like.
- Ensure this box is checked if you want this product to appear on your website.
- SKU stands for 'Stock Keeping Unit' which is used to store your own unique reference number for this product.
- Optionally enter a website address to link to from this product (eg. to link to a review of this product).
- Each product can be included in up to 2 different categories.
- If you had set up manufacturers and vendors, you would have the option to specify which ones this product belongs to here.
- This description is displayed when a user selects to view the details of the product (it does not appear in category listings).
- This description is displayed when the product is listed along with others in the same category.

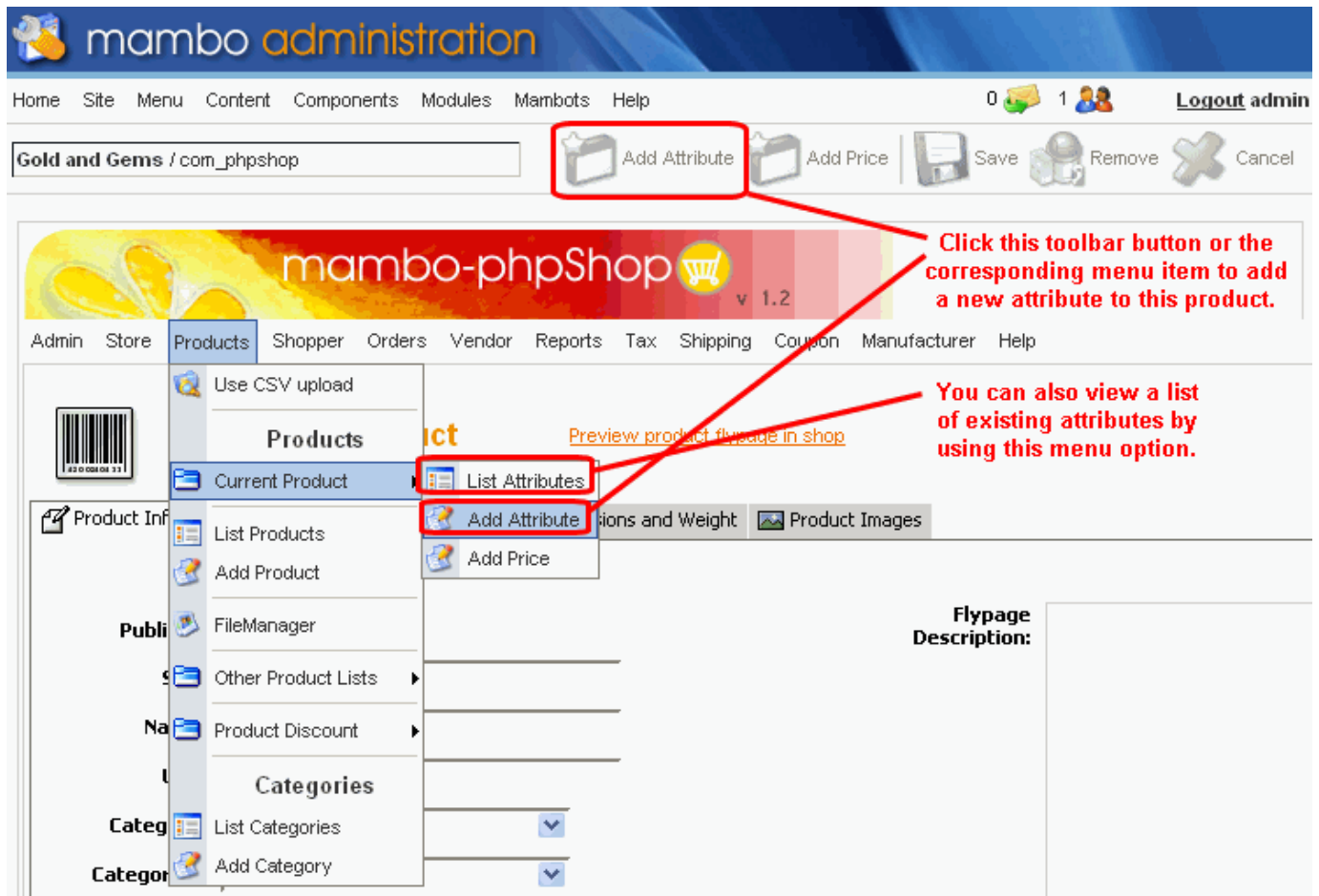
The next tab (**'Product Status'**) requires a little explanation. There are often times when you need to list variations on a product – for example, your product might come in different sizes, colours, shapes, etc. At times there may be optional extras which cost more, or even variations which cost less than the standard product offering.

There are 2 ways of presenting the variations on your basic product. You can either specify a list of configurations for the user to choose from, or split up the different attributes so that your user can create their own configuration. Or you can offer both choices if you like.

Either way, you need to specify the attributes that go into determining the configuration of your product variation. An attribute is just a name to which a list of allowed values can be assigned (for example an attribute name might be: 'colour', with the allowed values being 'red', 'green', and 'blue').

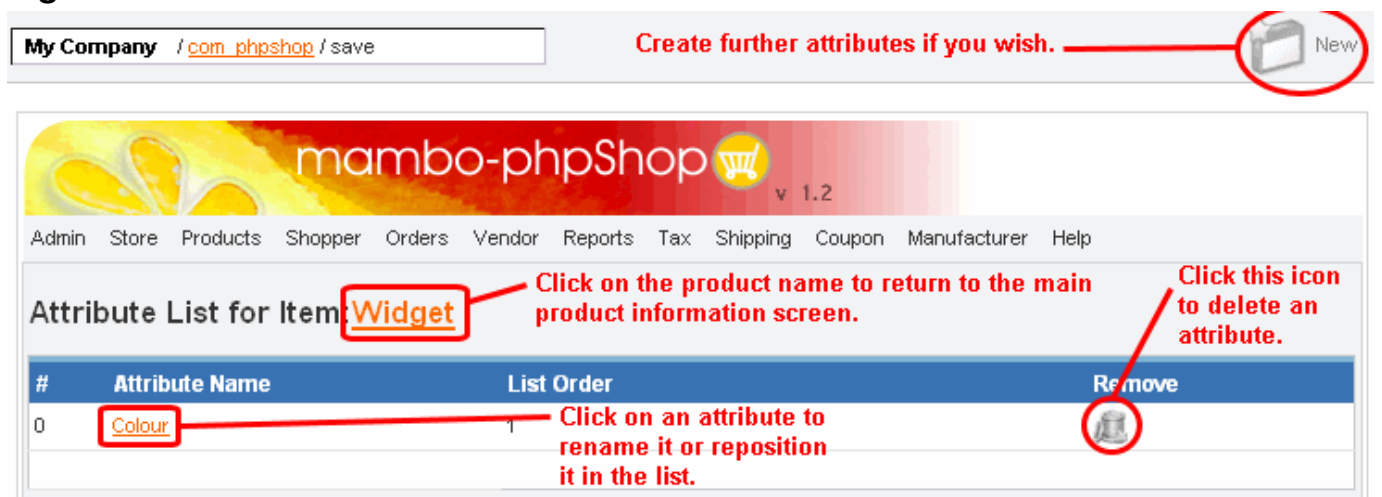
To add an attribute, click on the 'Add Attribute' toolbar button at the top of the screen (it doesn't matter which tab you are on in the product editor). See figure 25.

Figure 25 – Adding a New Attribute



You are then prompted for an attribute name and a 'list order'. List order is a numeric value which is used to sort the list of attributes into order – the same as on the shipping editor (figure 17). When you click on 'Save', you are taken to the main list of attributes for this product, which is illustrated in figure 26 (this is the same screen that you would get if you chose 'List Attributes' from the 'Current Product' sub-menu on the 'Products' menu, as shown in figure 25).

Figure 26 – The Attribute List



Now to specify the allowed values for each attribute, you need to return to the product editor, and switch onto the 'Product Status' tab (see figure 27).

Figure 27 – The Product Editor, Product Status Tab

Update Product [Preview product flypage in shop](#)

Product Information | **Product Status** | Product Dimensions and Weight | Product Images

In Stock: **You can use this field for stock control if you wish (the value will be updated automatically as products are ordered).**

Availability Date: **You can specify a date when this product will be available (click the elipsis button for a calendar).**

Availability: **Alternatively, specify a time period or waiting list - either by entering text in the text box, or selecting an image to convey the waiting period.**

On Special: **This option only has an effect if you install the 'Special Products' module - in which case, checking this box will cause this product to be featured.**

Discount Type: [You can use the Product Discount Form to add discounts!](#) **Ideally, this link would take you to the product discount form! Select 'Add/Edit Product Discount' from the 'Product Discount' sub-menu on the 'Products' menu to add discounts.**

Attribute List: **Enter attribute allowed values and prices in the format shown in the example. Note, you must first set up attributes using the 'Add Attribute' toolbar button.**

Examples for the Attribute List Format:

Size,XL[+1.99].M,S[-2.99];Colour,Red,Green,Yellow,ExpensiveColor[=24.00];AndSoOn...

Inline price adjustments for using the Advanced Attributes modification:

- + == Add this amount to the configured price.**
- == Subtract this amount from the configured price.**
- = == Set the product's price to this amount.**

Custom Attribute List: **If required, you can allow free text entry for additional fields which are not defined attributes (eg. if product can be personalised, you can allow the customer to enter the name or initials - just type the name of the extra field you want to add here, and a text box will be added to the site automatically).**

Examples for the Custom attribute List Format:

Name;Extras;...

So, using the example of a product which comes in a number of different colours, we could enter the following in the attribute list: `Colour,Black[-1.25],Red,Blue,Silver[+10.50],Gold[+25.00];`

This would mean that the 'Colour' attribute has allowed values of 'Black', 'Red', 'Blue', 'Silver', and 'Gold'. If the customer selects 'Black', the price of the product is reduced by £1.25 (assuming British currency here!). If Silver is selected, the price is increased by £10.50, and Gold attracts an extra £25 surcharge. Note that the attribute 'Colour' must be defined (using the 'Add Attribute' toolbar button) for this to work. You can set the allowed values of more than one attribute by separating the attributes with a semi-colon.

This method of presenting attributes allows the customer to set values for each attribute separately. In the shop's front end (ie. on your website), the product details page might look something like figure 28 (assuming 2 attributes have been defined: size and colour).

Figure 28 – Example of Attributes in the Front End

Widget

[Widget View](#) (Manufacturer)

[Full-Size Image](#) **Price:** £53.00

This widget is probably the most sophisticated widget currently on the market. It won 'best widget' award, 3 years running, and has been nominated as 'widget of the year' by the National Widgets Association (NWA).

Size:

Colour:

Individual attributes are presented for the customer to select any combination they want.

If you don't want your customer to mix-and-match attributes to their own desired configuration (for example, if your extra large widgets are only available in black), you can use an alternative method of display. We will discuss how to do this a bit later – first, let's finish off with setting up the product details. The next tab is the **'Dimensions and Weight'** tab, which is illustrated in figure 29.

Figure 29 – The Product Editor, Dimensions and Weight Tab

mambo-phpShop v 1.2

Admin Store Products Shopper Orders Vendor Reports Tax Shipping Coupon Manufacturer Help

Update Product [Preview product flypage in shop](#)

Product Information Product Status **Product Dimensions and Weight** Product Images

Product Dimensions and Weight

Length:

Width:

Height:

Unit of Measure:

Weight:

Unit of Measure:

Downloads

downloadable product?

EITHER Fill in a Filename:

OR Upload new File:

Self-explanatory, but make sure if filling in a file name, that you ONLY enter the NAME of the file (the path is specified in the DOWNLOADROOT field on the Downloads tab of the configuration screen).

If you are going to base your post and packing costs on weight or you want to indicate the size of the product, you can specify the values here. Values are optional, but if you do specify values, you must also specify units of measure.

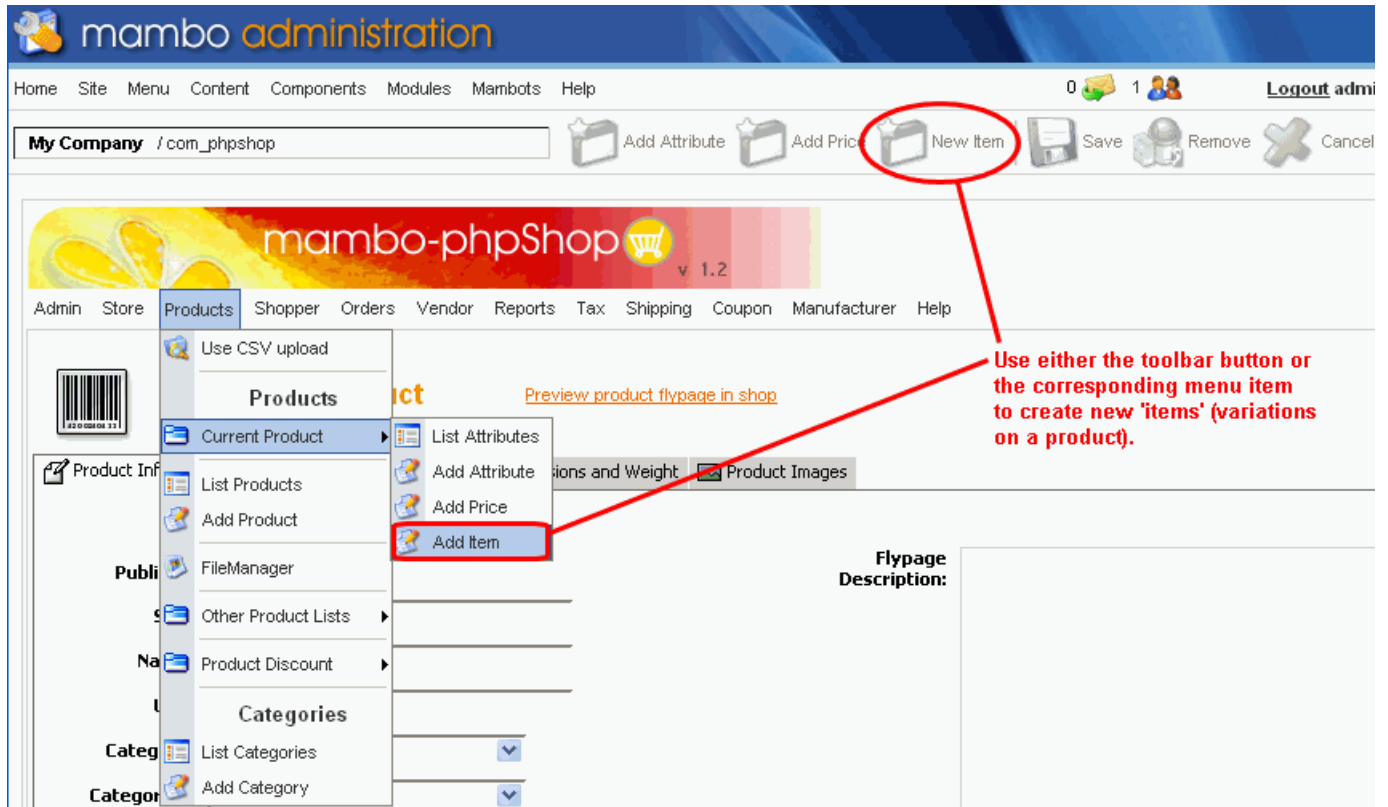
Finally, you can add images (thumbnail and normal-sized) for your product on the **'Product Images'** tab, shown in figure 30. Make sure you optimise your images so that the file sizes are nice and small – especially for the thumbnails.

Figure 30 – The Product Editor, Product Images Tab

The screenshot displays the 'mambo-phpShop v 1.2' interface. The top navigation bar includes 'Admin', 'Store', 'Products', 'Shopper', 'Orders', 'Vendor', 'Reports', 'Tax', 'Shipping', 'Coupon', 'Manufacturer', and 'Help'. The main content area is titled 'Update Product' with a link to 'Preview product flypage in shop'. Below this are tabs for 'Product Information', 'Product Status', 'Product Dimensions and Weight', and 'Product Images'. The 'Product Images' tab is active, showing two image upload sections. The 'Thumb Nail' section features a small image of a diamond ring, a text input field, a 'Browse...' button, and a 'URL: (Optional)' field. The 'Full Image' section features a larger image of the same ring in a jewelry box, a text input field, a 'Browse...' button, and a 'URL: (Optional)' field. A red box highlights the 'URL: (Optional)' field in both sections, with a red arrow pointing to a red text box on the left that reads: 'If you are hosting your images in separate webspace, you can link to them by entering the web address here instead of uploading.'

Going back now to offering variations on a product: If you want to offer a fixed list of variations for your customer to choose from, rather than allowing them to pick and mix attributes as they please, you need to create some **'items'**. Note: These are different from the content 'items' in Mambo itself. An item in mambo-phpShop is a pre-defined variation on the main product. In order to create an item, you have to set up at least one **attribute** first. Hence, it is only *after* you have added an attribute that the **'New Item'** toolbar button and its related menu option appear (as shown in figure 31).

Figure 31 – Adding a New Item



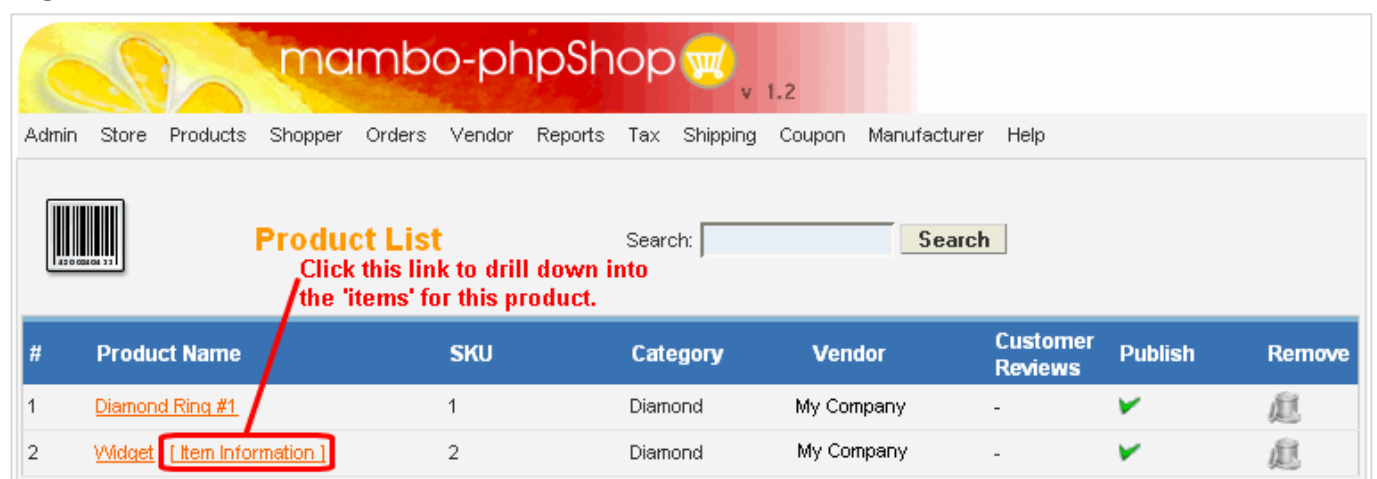
This is where it could get confusing! When you click on 'New Item' or 'Add Item', you are taken to a screen which looks identical to that shown in figure 24. You need to keep your wits about you so that you don't get lost – remember you are now dealing with a 'child' item, or variation of the main product.

You have all the same options as when adding the main product, and when a customer selects one of your items, the display of the product will switch to what you define for the item. Normally, you would probably only want to enter a new SKU and price, and perhaps a new image (for example to show the product in the chosen colour and size). But if you want to, you can enter a completely different name and description as well.

You can even add further **attributes** to the item itself – so a customer can select a variation of your product from a list, and then proceed to specify further attributes individually. You can also display *both* a list of fixed variations (items) *and* a list of the individual attributes for a product – perhaps to demonstrate the most popular combinations of attributes, or to apply a special price to certain combinations.

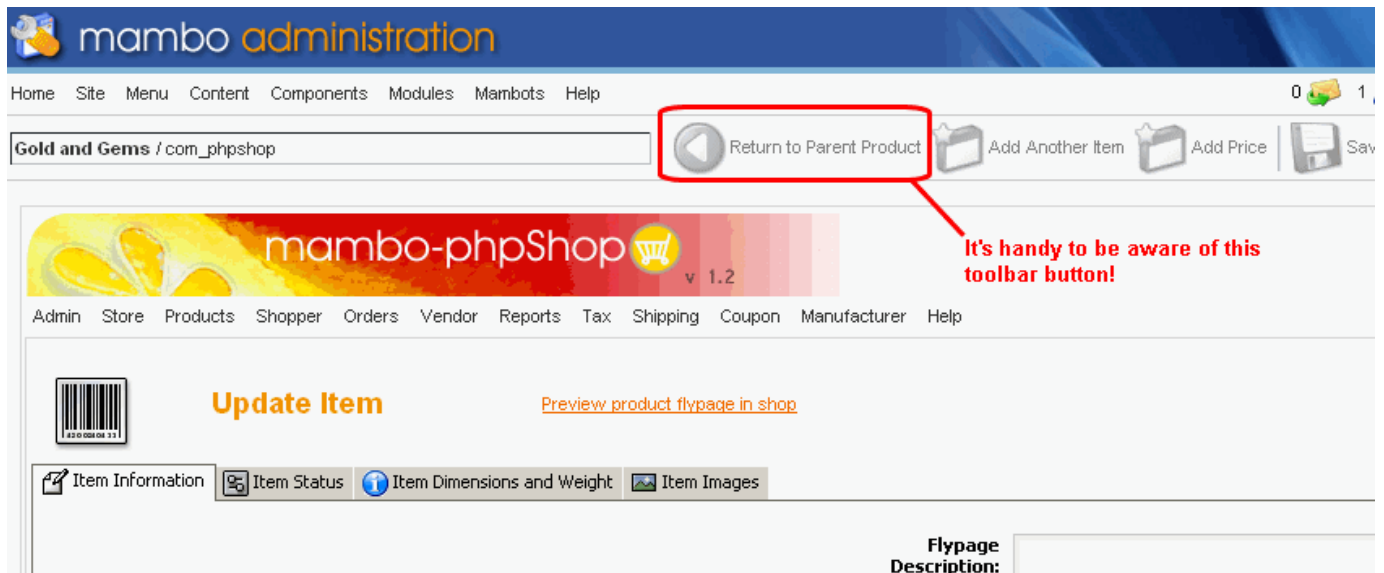
When you look at your list of products (like figure 23), if you have any items defined for a product, you will be able to 'drill-down' into them as shown in figure 32.

Figure 32 – Product List with Items



Another useful feature, is when you are looking at the details of an item, you can quickly jump to the editor for that item's **parent product** by using the 'Return to Parent Product' toolbar button (see figure 33).

Figure 33 – Return to Parent Product Toolbar Button



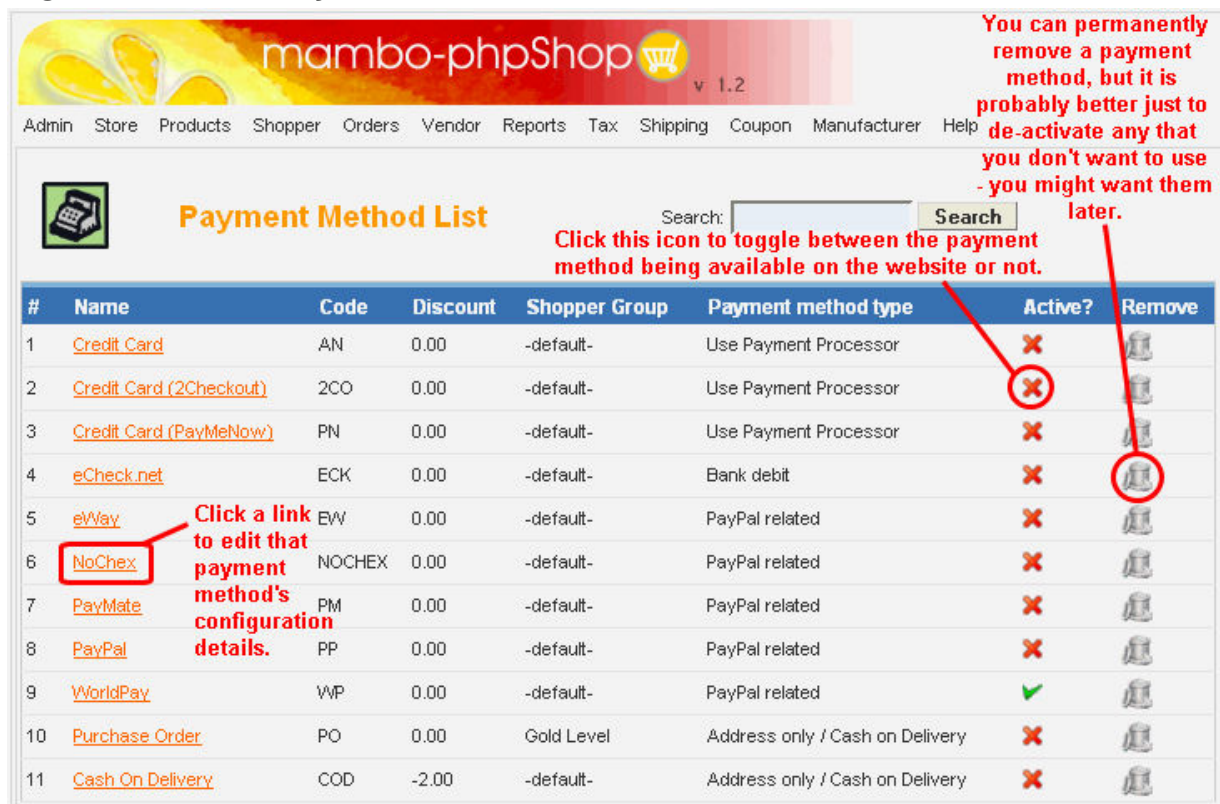
In the front end, your items will be displayed in a drop-down list so that the customer can select your pre-configured variation.

There are several other features which we have not really touched on (like adding separate prices for different types of shopper), but by now you are hopefully becoming familiar with the way things work, so it shouldn't be too difficult to work it out if you need to use some of the more advanced features. We still have some ground to cover though, so on we go to...

Payment Methods

The area of **payment processing** can get a bit complicated, and it might be best to leave this to your web designer to sort out if possible. If you are able to use one of the payment processing companies that are already supported by mambo-phpShop, it will make life a lot easier (you will need to sign-up with a payment processor yourself – the program won't do that for you!). You can see the options available by clicking on the **'List Payment Methods'** icon on the store summary, or by selecting 'List Payment Methods' from the 'Store' menu. See figure 34.

Figure 34 – The Payment Methods List



It is possible to create a new payment method (there is a 'New' toolbar button), but you may to hack the mambo-phpShop code in order to make use of it (not for the faint-hearted!)

The configuration options for each payment method are different – depending on what is required by that payment service provider. Usually your payment service provider will give you some kind of code, key, or user name which needs to be entered in the **Payment Method Editor**.

The Payment Method Editor consists of 2 tabs – the first of which you will probably not need to touch. Figure 35 shows the 2nd tab ('Configuration'), which varies depending on the payment service provider, and the example shown here is for WorldPay.

Figure 35 – The Payment Method Editor

The screenshot shows the 'Payment Method Form' configuration for WorldPay. The 'Configuration' tab is selected. A red box highlights the 'WorldPay Installation ID' field, with a callout stating: 'Your id code/user name/whatever as provided by your payment service provider (in this case WorldPay - who issue an 'Installation ID')'. Another red box highlights the 'Payment Extra Info' section, with a callout stating: 'The contents of this box depend on the requirements of the provider. In this case, WorldPay supply some sample form code which has been amended to work with mambo-phpShop - you may have to change this code when switching between test and live processing.' The code in the 'Payment Extra Info' section is as follows:

```
<form action="https://select.worldpay.com/wcc/purchase" method="post">
<input type="hidden" name="testMode" value="100">
<input type="hidden" name="instId" value="<?php echo WORLDPAY_INST_ID ?>" />
<input type="hidden" name="cartId" value="<?php echo $db->f("order_id") ?>" />
<input type="hidden" name="amount" value="<?php echo $db->f("order_total") ?>" />
<input type="hidden" name="currency" value="<?php echo $_SESSION['vendor_currency'] ?>" />
<input type="hidden" name="desc" value="Products" />
<input type="hidden" name="email" value="<?php echo $user->email ?>" />
<input type="hidden" name="address" value="<?php echo $user->address_1 ?>&#10<?php echo $user->address_2 ?>&#10<?php echo $user->city ?>&#10<?php echo $user->state ?>" />
<input type="hidden" name="name" value="<?php echo $user->title ?><?php echo $user->first_name ?>. <?php echo $user->middle_name ?><?php echo $user->last_name ?>" />
<input type="hidden" name="country" value="UK"/>
<input type="hidden" name="postcode" value="<?php echo $user->zip ?>" />
<input type="hidden" name="tel" value="<?php echo $user->phone_1 ?>" />
<input type="hidden" name="withDelivery" value="true">
<input type="hidden" name="fixContact" />
<br />
<input type="submit" value="PROCEED TO PAYMENT PAGE" />
</form>
```

Some payment processors will allow you to specify a script to be run on successful completion of a payment. Such a script could be used to **automatically update the order status** in mambo-phpShop so that you don't have to manually tie up all of your online receipts with your product orders. In the case of WorldPay, mambo-phpShop comes with a script to do this. The file is called '**worldpay_notify.php**', and it can be found in the following location: http://www.your-domain.com/your-mambo-folder-if-applicable/administrator/components/com_phpshop/. You will need to enter the appropriate URL (web address) for the script file in WorldPay's control panel.

Other payment service providers may use different methods of automatically notifying you when a payment is received. Again, it is probably easiest to get your web designer to sort out the requirements for this, but if you need to do it yourself, check with your payment service provider as to what their requirements are. Often, automatic notification is referred to as '**IPN**' or '**Instant Payment Notification**'.

Order Processing

Your shop is online, your catalogue is bursting at the seams, you've run all of your payment processing tests, and you're ready to go live! Well, maybe not quite. You need to know what to do when those orders come flooding in. When someone orders a product through your mambo-phpShop website, a record is created in the system with the details of the order.

The **order record** has a '**status**' to indicate where the order is in the stream of events – for example, whether it has been paid for, shipped, or cancelled. The actual values you use for status are up to you, but by default you are given status values of **Pending**, **Confirmed**, **Cancelled**, **Refunded**, and **Shipped**. If you want to define your own

status values, you can do so by selecting 'List Order Status Types' from the 'Order Status' sub-menu of the 'Orders' menu.

To view all of the orders in the system, click on 'List Orders' on the 'Orders' menu. This takes you to the Order List which is illustrated in figure 36.

Figure 36 – The Order Status List

The screenshot shows the 'mambo-phpShop v 1.2' interface. At the top, there is a navigation menu with items like Admin, Store, Products, Shopper, Orders, Vendor, Reports, Tax, Shipping, Coupon, Manufacturer, and Help. A red banner at the top right says: "You can optionally have mambo-phpShop send an e-mail to the customer to inform them that you have changed the status." Below the menu, the page title is "Order List". There is a search bar with a "Search" button. A status filter bar contains links: "Pending | Confirmed | Cancelled | Refunded | Shipped | List All". A table lists orders with columns: #, Order Number, Order Date, Last Modified, Status, Update, SubTotal, and Remove. The first row is highlighted. A dropdown menu is open for the first row's status, showing options: Confirmed, Pending, Cancelled, Refunded, and Shipped. Annotations with red boxes and arrows point to various elements: the order number "00000005", the status filter bar, the "Update Status" button, the "Remove" icon, and the dropdown menu.

Order List

Search:

[Pending](#) | [Confirmed](#) | [Cancelled](#) | [Refunded](#) | [Shipped](#) | [List All](#)

#	Order Number	Order Date	Last Modified	Status	Update	SubTotal	Remove
1	00000005	17-Jan-05 09:30	17-Jan-05 09:30	Pending	<input type="checkbox"/> Notify Customer? <input type="button" value="Update Status"/>	£5,386.62	
2	00000004	16-Jan-05 22:03	16-Jan-05 22:03	Pending	<input type="checkbox"/> Notify Customer? <input type="button" value="Update Status"/>	£5,387.45	
3	00000003	16-Jan-05 19:39	16-Jan-05 19:39	Pending	<input type="checkbox"/> Notify Customer? <input type="button" value="Update Status"/>	£3.68	
4	00000002	16-Jan-05 18:32	16-Jan-05 18:32	Pending	<input type="checkbox"/> Notify Customer? <input type="button" value="Update Status"/>	£5,386.62	
5	00000001	16-Jan-05 16:43	16-Jan-05 16:43	Confirmed	<input type="checkbox"/> Notify Customer? <input type="button" value="Update Status"/>	£16,152.51	

Copyright 2000 - 2004 Miro In... rights reserved.
Mambo is Free Software re... GPL License.
Page was gene... ds

When you click on an order number, you are given the full details of the order as shown in figure 37. You also have the ability to scroll through all of the order details using the 'prev' and 'next' links at the top of the page. The rest is up to you!

Figure 37 – The Order Details

The screenshot shows the 'mambo-phpShop v 1.2' interface. At the top is a navigation menu: Admin, Store, Products, Shopper, Orders, Vendor, Reports, Tax, Shipping, Coupon, Manufacturer, Help. Below this is a navigation bar with '< Prev | Next >' links. The main content is divided into sections: 'Purchase Order', 'Customer Information', 'Shipping Information', 'Order Items', and 'Payment Method'. Red annotations highlight specific features: the navigation links, the 'Update Status' button, the IP address field, the 'Customer's note' box, and the 'Order Items' table.

Purchase Order

Order Number: 00000002
 Order Date: 16-Jan-2005:18:32
 Order Status: **Pending** [Update Status]
 IP-ADDRESS: 127.0.0.1

Customer Information

Bill To	Ship To
Name: Mr. Test User	Name:
Company: Tests Unlimited	Company:
Address 1: 5 Nowhere Close	Address 1:
Address 2:	Address 2:
City: Testborough	City:
State/Province/Region:	State/Province/Region:
Zip/Postal Code: TE1 1ST	Zip/Postal Code:
Country: GBR	Country:
Phone:	Phone:
Fax:	Fax:
Email: test@netshine.co.uk	

Shipping Information

Carrier: Royal Mail
 Shipping Mode: Standard Delivery
 Shipping Price: £2.50

Customer's note

Order Items

Quantity	Name	SKU	Price	Total
1	Diamond Ring #1	1	£4,581.23	£4,581.23
1	Widget	2	£1.00	£1.00
SubTotal:				£4,582.23
Tax Total:				£801.89
Shipping and Handling Fee:				£2.50
Shipping Tax:				£0.00
Total:				£5,386.62

Payment Method

Payment Method	Account Name	Account Number	Expire Date
WorldPay			Dec-1999

Payment Log