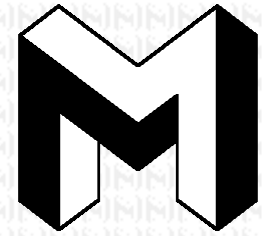


Currency Magic

Module for Miva Merchant

It's not easy being green. — Kermit the Frog



Copyright © 2006
by Magic Metal Productions

This document gives instructions on installing and using this module for Miva Merchant shopping-cart systems. If you need additional details, check with your Miva or hosting representative or documentation. You can also request help by email to support@TheMagicM.com.

This document applies to Currency Magic version 1.03 for Miva Merchant version 4.x, released February 2010. For Merchant 5.x, there is a separate set of files (module and documentation).

Currency Magic allows your store to display prices in multiple currencies. Customers to your site can simply pick the desired currency from a menu, and your entire Miva store will instantly start displaying all prices in that form!

- Exchange rates can be entered manually, or retrieved automatically from an on-line server.
- You can configure the module to retrieve exchange rates from many different servers. Upon installation, the module is preset to use Google.com. The documentation includes settings for several other public currency-converter sites, and explains how to create your own settings for other servers.
- Flexible formatting includes international punctuation selectable for each currency, and customized display of negative numbers.
- Advanced HTML and CSS formatting can be used, with plain text provided for emails and other unformatted displays.
- Supports OpenTokens for easy installation in all template systems.
- Compatible with all user interfaces (MMUI, OpenUI, templates).

Currency Magic does not require OpenUI, but it can use OpenUI tokens if they are available in your store. It is also compatible with the template systems from Copernicus and Viking Coders. If your store does not have any of these features (uses the standard Miva MMUI), you can still use Currency Magic (see "[Initial Setup](#)" below).

The screenshot shows a shopping cart interface with a yellow background. At the top right is a "Checkout" button. Below it is a table with columns for "Quantity", "Price/Ea.", and "Total".

Quantity	Price/Ea.	Total
2 <input type="text"/> <input type="button" value="Update"/>	¥436.43	¥872.87
3 <input type="text"/> <input type="button" value="Update"/>	¥8029.69	¥24089.07

Below the table, there is a discount line: "Discount on (2) 4-way Line Tester: ¥8029.69 cr". At the bottom right of the table area, a blue bar displays "Total: ¥16932.25".

To the right of the table is a "Choose your currency" dropdown menu. The menu is currently set to "Japanese Yen" and shows a list of other currencies: Brazilian Real, British Pound, Canadian Dollar, Chinese Yuan, Euro, Japanese Yen (highlighted), Swiss Franc, and US Dollar.

Customer advisory

This module is designed for use by all Miva Merchant users. Some understanding of HTML and Miva Script is needed if you want to customize the message displays or use tokens. However, the module can be used effectively without knowledge of HTML or tokens. When you install the module, all the settings are initialized with sufficient HTML for normal operation. So you should be able to get started using the module without writing any HTML or tokens.

Upgrading from an older version

NOTE: if you are installing the module for the first time, SKIP this upgrade information, and follow the instructions for "Installation" below.

The following steps are used to install a new version of the module on sites where it is already in use.

1. Log in to the Miva Merchant Administration page.
2. In the left-side menu, click the blue triangle next to the Modules link.
3. In the list of module links, find the one labeled **Currency Magic**, and click it.
4. In the right-side frame, click the Files link; then click the Upload button.
5. In the Upload box, click the Browse button.
6. Navigate to the folder where the module file is located, and select the file. The filename is `currencymagic.mvc`.
7. In the Upload box, check the Overwrite checkbox, and then click Upload. After a few moments, the Upload box will close.
8. In the right-side frame, click the Update button.
9. If you click the Information link, you should see that the version number is now 1.03.

Installation

The procedure for installing this module is the usual one for Miva Merchant modules: add the module to the site, and assign it to the store. The procedure is explained below in a bit more detail. If you need more information, see your Miva representative or documentation.

NOTE: If you intend to use this module with OpenUI or any template modules, you should install them before installing this module.

1. Log in to the Miva Merchant Administration page.
2. In the left-side menu, click the blue triangle next to the Modules link.
3. Click the Add Module link.
4. In the right-side frame, click the Upload button.
5. In the Upload box, click the Browse button.
6. Navigate to the folder where the module file, `currencymagic.mv` or `currencymagic.mvc`, is located, and select the file.
7. In the Upload box, click Upload. After a few moments, the Upload box will close.
8. In the right-side frame, click the Add button.

After the module is installed, you need to assign it to your store.

1. In the left-side menu, click the triangle next to the Stores link.
2. Click the link labeled with the name of your store.
3. In the right-side frame, click the link labeled Settings.
4. In the Currency drop-down menu, select Currency Magic.
5. Click the Update button in the lower right corner of the frame (you may need to scroll down to see it).

After the module is assigned to the store, a new "tab" link, labeled **Currency Magic**, will be seen at the top of the right-side frame. Clicking this link will take you to the module's settings screen.

Initial setup

After installing the module, you will need to take a few additional steps to get the currency menu onto the pages so customers can use it. How you do this depends on the type of user interface (UI) in your Miva store.

Module tokens

If you have a token-and-template UI, such as the Viking template modules or the Copernicus template engine, you must register Currency Magic to provide tokens. For the Viking modules, this is done by clicking the Third-Party Tokens link. For Copernicus or OpenUI, use the OpenTokens link on your "store page" (the admin page whose link is labeled with the name of the store). In either case, add a new token with module code `MMP_CURRENCYMAGIC`, and token code `CurMagic`.

After registering the module, you will place one or two tokens in appropriate locations in your store's page templates. There are two types of tokens:

- To generate the drop-down menu for selecting a currency, including HTML form and optional button, use the token `%module|CurMagic|menu|%`. If you place the menu in a location near the top or left side of the page, this may be the only token you need.
- If you place the menu token near the bottom or right side of the page, you may find that the module doesn't work correctly; when you select a new currency, the page display doesn't change. To correct this, add the token `%module|CurMagic|capture|%`, placing it near the top or left side of the page (the global header is a good choice). The capture token does not create any visible display; it just helps the module to retrieve the menu selection.

OpenUI without tokens

If you are using OpenUI without tokens ("hook points"), you don't need to do any additional configuration. You can select the location of the currency menu using the "Menu location" settings described [below](#).

Standard Miva (MMUI)

If you don't have any 3rd-party UI, you are using MMUI, the standard Miva Merchant UI. In this case, some custom programming may be required. Currency Magic can be used in a somewhat limited way without custom programming, by creating HTML forms and placing them in page headers and footers. For details on this, contact the developer at support@TheMagicM.com.

Overview

Currency Magic takes control of the places on Miva store pages where currencies are displayed: prices, shipping charges, discounts, etc. It changes the numbers, and the way they are formatted, based on the currency selection made by the customer. Note that **Currency Magic does NOT change the actual prices and other data in your store**. It just changes the way they are displayed. In the Miva admin pages, when you view orders in Order Processing, or edit your prices, shipping costs, etc., you will see the actual numbers that are stored in your databases.

Customers can change the display to their desired currency by simply making a selection on a drop-down menu. Admin settings and tokens allow you to put the currency menu in various places on any or all of your store pages.

Updating exchange rates

The module's admin page includes a table where you can enter information on various currencies, including the exchange rate from your home currency. You can enter the exchange rates yourself, or you can set the module to obtain rates by contacting a *rate server*: a Web site that distributes the data on-line. The module has admin settings that control how it retrieves rates; you can set it to obtain rate data from many different servers. Settings for several public servers are given later in this document. Upon initial installation, the module is configured to get rates from Google.com's currency page.

The module keeps track of the last date and time when each currency's rate was updated. Admin settings allow you to specify how long to wait before retrieving a new rate. For most users, updating every day, or even every week, is sufficient. You can select more frequent updates if you want, but there will be some processing overhead that may cause slight delays for your customers.

Display features

Besides monitoring exchange rates, Currency Magic also gives you control of how prices are displayed. You can apply HTML formatting to change the color, size, or other aspects of the appearance. You can customize how negative numbers are displayed. You can select different currency symbols and other punctuation for each currency.

A Miva store sometimes displays numbers in plain text, instead of HTML. Examples of this are the emails to customer and merchant, the costs displayed in the drop-down menu for selecting shipping, and various displays on the admin pages. Currency Magic provides plain-text formatting for these. Plain-text formatting uses the currency symbols and other punctuation that is defined for each currency, but removes HTML formatting that you may have set up for use on your store pages.

Module settings

To view the module's settings screen, go to your "store page:" in Miva admin, go to the Stores section of the left-side menu, and click the link labeled with the name of your store. Then click the **Currency Magic** link. The screen has three sections:

- Main settings and controls.
- Settings for controlling on-line updates of exchange rates.
- Settings for currencies that your store supports.

The various settings in each section are described below.

Main settings

This part of the screen displays settings that control the operation of the module and the appearance of various page elements.

Store internal currency

Select your own currency, the one that you use for your business operations. This should be the currency that was used to enter all the

data in the Miva databases, such as prices, shipping costs, etc. If your currency is not listed, you will need to add it to the currency table (see "[Currency settings](#)" below).

Default currency for display

Select the default currency for displaying prices to customers. This is usually the same as the internal currency, but not necessarily, e.g., if you have a U.S. business that sells mainly to customers in another country. The store will use this currency for new customers, or returning customers whose session has expired. Customers can override the default by making a selection on the currency menu. The module will remember the selection as long as the customer's session remains active.

Formatting for prices

These two text boxes let you enter text or HTML tags that are used to enclose all prices that the module displays. For example, you can put a `` tag in the Before box, and a `` tag in the After box, to change the color, size, typeface, etc.

Tip: Although sans-serif fonts such as Arial or Helvetica are generally easier to read on computer screens, a serif font such as Times may look a little more elegant or professional. Also, making the prices a little larger than surrounding text can make them more readable.

Tip: Instead of using formatting tags such as ``, you may be able to use tags with `class` attributes, and apply the actual formatting with CSS style sheets. This would allow currency displays to have a different appearance on various pages, instead of being the same throughout the store.

NOTE: This formatting is not applied to plain-text prices such as those in the emails to the customer and merchant.

Formatting for negative numbers

Businesses use many different ways to display negative numbers, such as:

- A minus sign before the number.
- The letters "CR" (for credit) after the number.
- Enclosing the number in parentheses.
- Displaying the number in red, or some other color.

Currency Magic can do any of these, and more. The Before and After text boxes allow you to enter text or HTML that will be placed around any prices that are negative. You can simply enter characters, such as a minus sign in the Before box, or opening and closing parentheses in both boxes. You can also enter HTML tags to change the color, or apply other formatting such as italics or underlining.

Some of the numbers displayed by the Miva store must be in plain-text form, not HTML; so Currency Magic has two sets of text boxes for negative numbers. You can use one format in HTML pages, and another, simpler one in plain text.

HTML for currency menu

Currency Magic displays a drop-down menu and optional button that customers use to select their desired currency. You can customize the appearance of this menu by entering HTML, text, and one or two special tokens in this text box. At a minimum, the text box must contain the `%menu%` token, which represents the drop-down menu. You may also enter the `%button%` token to specify the location of a button to trigger the currency selection. If you do not use a button, the drop-down menu will automatically trigger a currency change whenever a customer makes a selection. You can also add other text such as a label or caption for the menu, enclose it with `<table>` or `<div>` tags to give it a colored border, etc.

If your store has a recent version of OpenUI, you can use OpenUI tokens in the HTML. As usual, if you want to do this, you must put the special token `%OUI%` at the very beginning of the text.

HTML for menu button

If you have included a `%button%` token in the currency menu, you enter HTML code for your button in this text box. The button can be as simple as:

Formatting for prices	Before	<input type="text" value="<fontface='Times' size='+1'></>"/>
	After	<input type="text" value=""/>
HTML for negative numbers	Before	<input type="text" value=""/>
	After	<input type="text" value="cr"/>
Plain text for negative numbers	Before	<input type="text"/>
	After	<input type="text" value="cr"/>
HTML for currency menu	Tokens: %menu% %button%	
	<input type="text" value="<div align='right'>
<table cellpadding='0' cellspacing='6'><tr><td bgcolor='#008000' align='center'>Choose your currency<input type='button' value='Choose'</td></tr></table></div>"/>	
HTML for menu button	<input type="text" value="<input type='submit' value='Choose'>"/>	
Location of menu	<input type="radio"/> None (using tokens) <input type="radio"/> Top of page body <input type="radio"/> Under global header <input checked="" type="radio"/> Bottom of page body <input type="radio"/> Top of left-side menu <input type="radio"/> Over global footer	

<input type="submit" value="Choose">

You can type some other label text instead of Choose. You can also use other types of buttons, such as images or hyperlinks with JavaScript.

Location of menu

If you are using OpenUI without tokens (the older "hook points" system), these buttons allow you to select the location where the currency menu appears on your store pages. If you **are** using tokens, select None.

On-line update settings

These settings control how the module communicates with rate servers to obtain exchange rates.

Enable

Check this box to allow on-line updating of exchange rates. If this box is unchecked, the currently saved rates will remain fixed, unless you change them manually (see below).

Update interval

These settings let you select how often the module will check the rate server. Too-frequent checking may slow down your own site's response time. Once a day, or even once every 7 days, is probably sufficient for most stores. If the financial markets become volatile, or if your business is especially sensitive to changes in rates, you could choose an interval measured in hours or even minutes.

NOTE: A setting of 0 in the text box will cause the module to check the exchange rates **every** time a customer views a page that includes the currency menu. This is intended only for testing.

On-line updates

Enable

Update interval days (Enter 0 for immediate update — for testing only)

Update URL

Retrieval pattern Enter count type = target (one per line)

```
1 text = 1 %fromcode% =
1 text = | %tocode%
```

Manual retrieval Enter currency code and click Update.

Update URL

Enter the URL of the rate server. URLs for several public sites are listed in the table below. Note that the URL may contain special tokens that are converted to currency codes or names. Some examples of this are given below.

Retrieval pattern

The retrieval pattern is a set of instructions that tells the module how to find the rate within the data sent from the rate server. Patterns for several public sites are listed in the table below. A detailed description of the patterns is given [later in this document](#).

Manual retrieval

When adding a new currency to your store, you may want to immediately trigger retrieval of the current exchange rate. You can do that by entering the currency code in this text box, and then clicking the Update button at the bottom of the page. This can also be useful when testing a new retrieval pattern, or to check the module's operation after changing any settings.

Example settings

When Currency Magic is installed, it is initially configured to retrieve exchange rates from Google.com's financial page. The following table lists settings that you can use to retrieve exchange rates from several public Web sites.

Server	Settings
Google	URL: <code>http://www.google.com/finance?q=CURRENCY:%fromcode%%tocode%</code> Retrieval pattern: <code>1 text = 1 %fromcode% =</code> <code>1 text = %tocode%</code>

CNN Money.com	URL: <code>http://money.cnn.com/data/currencies/index.html</code> Retrieval pattern: 1 text = %todesc% 2 tag = /td
GoCurrency.com	URL: <code>http://gocurrency.com/v2/table_f.php?cur=%fromcode%;%tocode%&v=1</code> Retrieval pattern: 2 tag = tr 4 tag = /td

Currency settings

The module displays a table listing a number of currencies, with their code, exchange rate, and other settings that control how they are displayed. You can change any of these settings by entering new values in the table and clicking the Update button at the bottom of the page.

Code	Description	Prefix	Suffix	Spacer	Decimal	Rate	Show	Auto-update	Last update	Remove
BRL	Brazilian Real	R\$.	.	2.152	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Apr. 11, 2008 21:55	<input type="checkbox"/>
GBP	British Pound	£		.	.	0.573	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Apr. 11, 2008 21:55	<input type="checkbox"/>
CAD	Canadian Dollar	\$	CN	.	.	1.149	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Apr. 10, 2006 21:46	<input type="checkbox"/>
JPY	Japanese Yen	¥		.	.	118.300	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Apr. 11, 2008 21:55	<input type="checkbox"/>
CHF	Swiss Franc	SFr.		'	.	1.301	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Apr. 11, 2008 21:55	<input type="checkbox"/>
USD	US Dollar	\$	US	.	.	1.000	<input checked="" type="checkbox"/>	—	—	—
							<input type="checkbox"/>	<input type="checkbox"/>	New currency	

Upon installation, the module will support a number of common currencies. You can add more by entering the necessary information in the New Currency boxes at the bottom of the table, and clicking Update. The table is sorted in alphabetical order by description; so even though you add new currencies at the bottom of the table, once you click Update, they will move to their proper location.

NOTE: You can include HTML formatting in the text boxes described below, but it will **not** be removed when generating plain-text numbers. This may cause a problem in parts of your store that use plain text, such as the emails that your store sends, or the drop-down menu for shipping costs.

Code

There is a standard set of codes (ISO 4217) to identify currencies. A complete list of codes can be obtained at various Web sites, including:

- <http://www.xe.com/ict/>
- <http://www.trigeminal.com/samples/setlocalesample2.asp> – this site also lists the currency-symbol characters and other punctuation usage for each currency.

Tip: If you just want to look up a few codes, you can go to one of the public converter pages, such as <http://www.oanda.com/currency/converter/>, and simply scroll through the drop-down menu until you find the country you're looking for.

Description

Enter a name or description for the currency, such as "Dutch Guilder" or "Zambian Kwacha." This will be used on the currency menu to identify the currency to customers.

Prefix and suffix

You can place a currency symbol or other markers either before or after the number; various countries do it both ways.

Tip: Canada, Bermuda, and many other countries use the dollar sign prefix for their currency. You can use a suffix such as (CA) to distinguish each country.

Tip: You can put a space character at the end of the prefix, or the start of the suffix, to leave a bit of space between the

price and the punctuation.

Spacer character

For long numbers, the module can enter a spacer character after every three digits. Not all countries use a comma for this; many use the period for this, and a few use other characters. Enter the desired character in this box.

Decimal character

Not all countries use a period as a decimal point; many use the comma. Enter the desired character in this box.

Rate

This text box shows the exchange rate for the currency, compared to your store's internal currency. For example, if your internal currency is US dollars, this text box will tell you how many of the other currency is equivalent to one dollar. If auto-updating is enabled for the currency, the box will be filled in automatically the next time a customer views a page with the currency menu. You can also trigger a manual update, or simply type in a number.

NOTE: if the rate is empty or zero, the currency will not be included on the currency menu for customers. If auto-updating is enabled, then whenever a customer views the currency menu, the module will try to retrieve new data for any currency that has a zero rate.

Show

If this box is checked, the currency will be included on the currency menu for customers (unless the exchange rate is zero, as described above). You can uncheck this box in order to remove a currency from the store without deleting it from the table.

Auto-update

If this box is checked, the module will update the exchange rate by contacting the rate server as needed, based on the selected update interval.

Last update

The module remembers the latest time and date when the rate was updated, either by on-line retrieval or by someone manually entering a new number. This date is displayed in red if the rate has "expired," i.e., if it is due for a new update, based on the stored date/time and the update interval setting.

"Expired" rates are still used in the store; a currency is **not** removed from the menu simply because the rate has not been updated recently. Sometimes Internet glitches may interfere with on-line updates. If this happens, the module will try again the next time a customer views the currency menu. In the meantime, the module will use the old exchange rate.

Remove

To remove a currency from the store, check this box and click the Update button at the bottom of the page.

On-line retrieval patterns

The module obtains on-line updates by sending a request to the update URL, which may be a public Web page that contains a lot of other text, images, advertising, etc., as well as the actual rate data that you want. The retrieval pattern tells the module how to sort through all the other text to find the rate. It's a set of instructions that tell the module to look for specific "clues" in the retrieved HTML, which eventually lead to the exchange rate.

Writing these patterns requires that you have some understanding of HTML. You need to be able to read the source of a Web page, and understand the arrangement of tags and text, in order to figure out a description for the point where the exchange-rate data is located. Creating patterns is mainly a task for developers and power users. This document gives enough information for skilled users to create their own patterns, or to modify the existing ones, in case of changes in the format of a rate-server page.

The pattern consists of one or more lines of text. Each line contains three elements: a count, a type, and a target. There must be a space after the count, and an equal sign after the type. Other spaces and tabs may be added for clarity.

Patterns can be explained with a simple example. Suppose you want to retrieve exchange rates from a Web site where the page consists of several HTML tables. After studying the layout, you find that the rate is contained in the 3rd column and 2nd row of a table, which is just below a subtitle that reads "Your exchange rate." The retrieval pattern for this would be:

```
1 text = Your exchange rate
1 tag  = table
2 tag  = tr
3 tag  = /td
```

The first line instructs the module to find the first occurrence of the text, "Your exchange rate." The second line tells the module to look further, until it finds a `<table>` tag. The third line says to proceed from there to the **second** occurrence of a `<tr>` tag (because of the number 2). This brings the module to the second row of the table. The fourth line says to search on for the **third** occurrence of a `</td>` tag.

When the search is completed, since the pattern ended with a tag step, the module retrieves the last piece of text that it read **before** finding the final search target. That's why the last line of the pattern specifies the `</td>` tag, instead of `<td>`. If the pattern ended with the third `<td>`, the desired text would not have been read yet.

Another way to retrieve an exchange rate is with a pattern that ends with a `text` instruction. In this case, you add the vertical-bar character "|" to the target. This character serves as a marker to identify the point in the text where the rate will be found. This is described in more detail in the next section.

Here are some other notes on creating patterns.

- When processing a `text` instruction, the module removes white-space characters such as spaces and Newlines from the beginning and end of the retrieved text; so you don't have to put these characters in your patterns. (The module only removes white space from the beginning and end, not from inside the text.)
- `text` instructions can only find their target if it matches **all** the text between two tags, except for white space characters as described above. For example, if a Web page includes the phrase "your data," you can't just search for the word "data." Also, `text` instructions can not find a piece of text with any HTML tags inside it; for example, you may **not** write a `text` instruction with a target such as `Here's the latest data.`

Tokens for patterns

Many rate servers include the code or name of the currency on the page that displays the exchange rate. It's helpful to write patterns that search for this; but you can't just type the code or name into the pattern, because the same pattern is used for retrieving all currencies. The solution is to use a special token, which allows the pattern to find text that varies according to the currency being updated.

- The token `%fromcode%` represents the currency code of your store internal currency, as defined in the table (see "[Currency settings](#)" above).
- The token `%fromdesc%` represents the description of your store internal currency.
- The token `%tocode%` represents the currency code of the currency being updated.
- The token `%todesc%` represents the description of the currency being updated.

Also, the vertical-bar character `|` is used as a marker to identify the point in the retrieved text where the rate will be found. Note that this marker is used only in `text` instructions.

This marker is used in the pattern for Google shown above:

```
1 text = 1 %fromcode% =  
1 text = | %tocode%
```

Google provides text including both currency codes, such as **1 USD = 90.09 JPY**. The second line of the pattern has a target that consists of a vertical-bar character, followed by a space and the token `%tocode%`. This tells the module to look for a piece of text that starts with the rate itself (represented by the vertical bar) followed by a space and the currency code.

The sample patterns in the table [above](#) show several other uses of these tokens.
