Developer Manual

xFlow v5.x

Created by:
Envex Developments Inc.
http://www.envex.com/

Version 1.2
April 24th, 2005

Table of Contents

1.) Introduction

This manual is meant for HTML developers, and those responsible for the technical aspects of xFlow. Many things are explained in this manual, such as how to modify HTML templates to customize the look, how to revert the database to a backup, how to transfer the database to another server, plus more.

2.) HTML Templates – Overview

xFlow uses HTML templates, allowing you to customize the look of every aspect of the software. All HTML templates are stored in the */xflow_data/htmlt/* directory of your web server, and the only ones that you need to be concerned with are those located in the */xflow_data/htmlt/marea/* directory.

An HTML template is nothing more than a simple web page, that contains some special HTML tags. Below describes all special HTML tags used within the templates:

2.1 Merge Fields

The HTML templates contain many merge fields, which are replaced with different information when displayed. All merge fields are surrounded with **~** marks, and should never be modified. Some examples of merge fields are, *~username~, ~total_downline~*, and so on.

2.2 <*REPRODUCE*> HTML tags

Throughout the HTML templates, you will you notice several sets of <REPRODUCE ... > HTML tags. When the software prints out an HTML template , it will take everything between these HTML tags and copy it over and over. For example, if the software is displaying a list of members, the HTML template will contain one set of <REPRODUCE ... > HTML tags, and the software will copy everything between the HTML tags over and over, for every member.

You may edit the HTML code inside of these HTML tags, but should never delete or modify the actual HTML tags themselves. Only the HTML code inside of them.

2.3 <*IF*> HTML tags

Another special set of HTML tags which you will see in many HTML templates are the <IF ... > tags. Depending on how your program is setup, the script may or may not display the contents between these tags when it prints an HTML template. For example, several HTML tags contain special sections for multiple program structures. If your program contains multiple program structures, the software will display the contents between the <IF ... > tags. However, if your program does not contain multiple program structures, the software will "hide" the contents between these tags.

Again, you may modify the contents between these tags, but should never modify or delete the HTML tags themselves.

2.4 <*INCLUDE*> HTML tags

The <INCLUDE ... > HTML tags are quite rare in xFlow. These simply act exactly as Server Side Includes do. The <INCLUDE ... > HTML tag will contain the location of another HTML template. When the software prints the HTML template it will simply replace the <INCLUDE ... > HTML tag with the contents of the specified template. You should not modify these HTML templates at all, and they should remain exactly as they are.

3.) HTML Templates – Descriptions

The below table lists all templates located in the */xflow_data/htmlt/marea/* directory, and their use. You can edit these HTML templates just like normal web pages, but please keep in mind the special HTML tags explained in *Section 2. HTML Templates – Overview*.

HTML Template	Description
account_summary.htmlt	When a member views their account summary from the Member's Only Area. This is basically the same as viewing a member through the *Database->View Member* menu of the Admin Control Panel.
delete_account.htmlt	When a member wants to delete themselves through the Member's Only Area. This template simply asks the member to confirm their deletion, by entering "CONFIRM" in a text box.
delete_account_success.htmlt	After a member has successfully deleted their account from within the Member's Only Area, this template will be displayed.
deposit_funds.htmlt	When a member wants to deposit funds into their account. This template only asks for the amount they would like to deposit.
edit_profile.htmlt	When a member edits their profile through the Member's Only Area.
email_downline.htmlt	When a member sends an e-mail message to their downline. This template asks for the contents of the e-mail message to send.
email_downline_preview.htmlt	Before actually sending the e-mail message to a member's downline, the member must first preview their message. This template shows a preview of the e-mail message, asking the member to confirm it.
login_expired.htmlt	If a member tries to navigate through the Member's Only Area, after being inactive for too long, thus being automatically logged out, this template is displayed.
login_invalid.htmlt	If a member enters an invalid username or password when trying to login to the Member's Only Area, this template is displayed.
logout.html	After a member has successfully logged out of the Member's Only Area, this template is displayed.
payment_info.htmlt	When a member wants to manage their payment information. This shows a list of all deposit and withdraw payment methods on the member's account, allowing them to change the default payment method.

payment_info_add.htmlt	When a member wants to add a new payment method to their account. This template asks for the payment information.
process_transaction.htmlt	When a member processes a specific transaction on their account. This template gives a summary of the transaction details, and asks the member to confirm it.
process_transaction_ccinfo.htmlt	Used within an *<INCLUDE>* tag within the *process_transaction.htmlt* template, allowing a member to enter new credit card information.
processing.htmlt	Currently only used when sending an e-mail message to a downline. This template shows the blue progress bar, as the e-mail is being sent.
shop_add_success.htmlt	Displayed after a member successfully adds a product into their Shopping Cart from within the Member's Only Area.
shop_browse_category.htmlt	Displayed when a member browses all categories and products within the Shopping Cart from the Member's Only Area.
shop_checkout.htmlt	Displayed when a member checks out, and goes to submit payment for products purchased from the Shopping Cart, through the Member's Only Area.
shop_view_cart.htmlt	Displayed when a member views and manages their individual Shopping Cart from within the Member's Only Area.
shop_view_product.htmlt	Displayed when a member views details on a product from the Shopping Cart through the Member's Only Area.
success.htmlt	Every time a member successfully completes an action, such as editing their profile, this template is displayed.
transaction_approved.htmlt	When a member processes a transaction on their account, which is approved, this template is displayed.
transaction_declined.htmlt	When a member tries to process a transaction on their account, and payment is declined, this template is displayed.
transaction_error.htmlt	When a member tries to process a transaction on their account, and an error occurs when processing the payment through the merchant account, this template is displayed.
transaction_history.htmlt	When a member views their transaction history through the Member's Only Area. Same as the *Transaction->Manage Account* menu through the

	Admin Control Panel.
transaction_history_month.htmlt	When a member views a list of all transactions on their account, for any given month. This template is displayed after a member clicks on a specific month, from the transaction_history.htmlt template.
transaction_payment_ach_us.htmlt	A small template, which is never displayed by itself. Instead, it's used within *<INCLUDE>* tags in other templates, and displays the ACH bank account information used for transactions, if needed.
transaction_payment_ach_us_wtd.htmlt	A small template, which is never displayed by itself. Instead, it's used within *<INCLUDE>* tags in other templates, and displays the ACH bank account information used for withdrawal transactions, if needed.
transaction_payment_cc.htmlt	A small template, which is never displayed by itself. Instead, it's used within *<INCLUDE>* tags in other templates, and displays the credit card information used for transactions, if needed.
transaction_view.htmlt	When a member views details of any one transaction on their account. This template is displayed when a member clicks on a specific transaction, from the transaction_history_month.htmlt template.
upload_image.htmlt	Displayed when a member uploads a new profile image to be displayed on their self replicating web site.
upload_image_success.htmlt	Displayed as a success message, after a member successfully uploads a new profile image.
view_downline.htmlt	When a member views their downline through the Member's Only Area. Same as the *Geneology->Manage Account* menu of the Admin Control Panel.
view_downline_level.htmlt	When a member views the member's on a specific downline. Displayed when a member clicks on a "More Info" link from the view_downline.htmlt template.
view_downline_level_rep.htmlt	An *<INCLUDE>* template, used by the view_downline_level.htmlt template. This template contains the table row, which is replicated over and over, for each downline member to display.
view_downline_rep_upline.htmlt	An *<INCLUDE>* template, used by the view_downline.htmlt template. This template contains one table which, which is replicated over and over, for each upline member to display.
view_downline_rep2.htmlt	An *<INCLUDE>* template, used by the

	view_downline.htmlt template. This template is replicated over and over, for every downline position the member has.
view_downline_rep3.htmlt	An *<INCLUDE>* template, used by the view_downline_rep2.htmlt template. This template contains one table row, and is replicated over and over, for every downline level the member has.
view_downline_tree.htmlt	Displayed when a member views their downline in a tree format, and has more than one downline position in forced matrix program structures. This template asks which downline position they would like to view.
view_downline_tree2.htmlt	Displayed when the member views their downline in a tree format. This template shows the full downline in the actual format.
view_website_stats.htmlt	When a member views their web site statistics through the Member's Only Area. The same as viewing a member's web site statistics through the *Database->View Member* menu of the Admin Control Panel.
withdraw_funds.htmlt	When a member wants to request a funds withdrawal from the Member's Only Area. This template asks the member how much they would like to withdraw, from their account.
withdraw_funds_success.htmlt	After a member has completed the withdraw_funds.htmlt template, and requested a funds withdrawal, this template is displayed, telling the member their request has been successfully received.

There are also several templates located in the */xflow_data/htmlt/* directory, which you may want to edit. A description of all templates located in the */xflow_data/htmlt/* directory are explained in the below table.

HTML Template	Description
dblock.htmlt	Displayed when the software tries to access the database, while the database is locked. For example, if the software is performing a backup of the database, this template will be displayed when someone tries to do anything, which requires accessing the database.
deactivated.htmlt	Displayed when someone tries to view a member's self replicating web site, who has been deactivated.
error.htmlt	Displayed every time any error occurs in the Member's Only Area, or through any public action, such as joining.
error_dbconnect.htmlt	Displayed when, for some reason, the software is unable to connect to the database. This template

	should never be modified.
error_mail.htmlt	If an error ever occurs when trying to send an e-mail, a small browser window will automatically open, and display this template, which lists all e-mail address the software couldn't e-mail.
error_priveledge.htmlt	Displayed when a member tries to perform an action through the Member's Only Area, which isn't permitted, due to the priviledge settings you setup through the *Setup->Priviledge Settings* menu of the Admin Control Panel.
newuser_thankyou.htmlt	Displayed after a new member successfully filled out the newuser HTML form, and joins your program.
reminder.htmlt	When someone clicks on the "Forgot your password" link. This template asks for the member's e-mail address, and will e-mail them their password.
reminder_success.htmlt	After someone successfully completes the reminder.htmlt form, and their password has been e-mailed, this template is displayed.

3.1 Edit Profile

As you may notice, the *edit_profile.htmlt* template seems to be very restrictive, due to the *<REPRODUCE>* HTML tags. If wanted, you can completely customize the look of the edit profile form. It's not absolutely required that you actually use the *edit_profile.htmlt* template. Instead, you can create your own web page, upload it to the Member's Only Area, and use that to allow member's to edit their profile. Below explains exactly how to create your own edit profile form.

First, you must have a *<FORM>* tag, which will be:

```
<form action="index.cgi?~authtext~&action=edit_profile" method="POST">
```

Next, you need to add the text fields to your form, which contain the member's profile, allowing members to edit their information. Every text field should look something like:

```
<input type="text" name="field" value="~field~">
```

Where *field* is the name of any field in your member database, except 'username'. Please note, all fields you add to the form, must also be checked in the *Setup->General Settings* menu, allowing members to edit those fields. For example, if you wanted to allow members to edit the "email" field from your member database, you would add the following text field:

```
<input type="text" name="email" value="~email~">
```

Last, you need to add a submit button to the form, such as:

```
<input type="submit" value="Edit Profile">
```

Other than the above mentioned items, you can create the edit profile form in any way you wish. Once you've created the new HTML form, save it as a web page, and upload it to your

Member’s Only Area through the *Setup->Manage Web Site* menu of the Admin Control Panel. Then, just link to this web page, to allow members to edit their profile.

3.2 Process Transactions

One of the features of xFlow is members can process transactions through the Member’s Only Area, which you create through the *Transaction->Transaction Settings* menu of the Admin Control Panel. This is done by using the *process_transaction.htmlt* HTML template, but at times, you may not want to use this template.

If wanted, you can create your own web page(s) to process transactions, and are not required to use the *process_transaction.htmlt* template. Below explains exactly how to create an HTML form, which you can upload to your Member’s Only Area, and allow members to process a specific transaction.

First, you must have a *<FORM>* tag, which will be:

```
<form action=”index.cgi?~authtext~&action=process_transaction”
method=”POST”>
```

Next, you need a hidden form field, which gives the transaction ID# you would like to process. To find the transaction ID#, go to the *Transaction->Transaction Settings* menu of the Admin Control Panel, and click on the needed “View Details” link to find the appropriate transaction. The next table which is displayed, will show the ID# of all transactions. You need to add the following hidden field into the HTML form:

```
<input type=”hidden” name=”transaction_id” value=”XXX”>
```

Of course, replace *XXX* with the transaction ID# you would like to process. If wanted, you may also place a text field, allowing members to enter a coupon code, by placing the following field into your form:

```
<input type=”text” name=”coupon”>
```

Otherwise, if you would rather not allow members to enter a coupon code, you must add a hidden form field to your form, which is:

```
<input type=”hidden” name=”coupon” value=””>
```

Last, the only other requirement is to place in a submit button in the form, such as:

```
<input type=”submit” value=”Process Transaction”>
```

Other than that, the HTML form to process new transactions can be created in any way you wish. It only needs to conform to the few simple requirements listed above.

3.3 Delete Account

If for some reason you do not wish to use the *delete_account.htmlt* template, you may create your own web page, allowing members to delete their account, by completing the following:

First, your HTML form needs to *<FORM>* tag, which is:

```
<form action=”index.cgi?~authtext~&action=delete_account” method=”POST”>
```

The only other requirement is one text field, which the member must type “CONFIRM” into, to confirm their account deletion. This text field must be:

```
<input type="text" name="confirm">
```

If you would rather not require members to confirm their account deletion, and instead, only need to click on a submit button, you can use a hidden form field instead, such as:

```
<input type="hidden" name="confirm" value="CONFIRM">
```

Last, you simply need to add a submit button to the form, such as:

```
<input type="submit" value="Delete Account">
```

3.4 Deposit Funds

For some reason, you may also not want to use the *deposit_funds.htmlt* template, allowing members to deposit funds into their account. You can create your own web page, allowing members to deposit funds into their account, by completing the following:

First, you need a *<FORM>* tag, which will be:

```
<form action="index.cgi?~authtext~&action=deposit_funds" method="POST">
```

The only required text field is to allow members to enter in the amount, they would like to deposit, and will be:

```
<input type="text" name="amount">
```

Then the member's default deposit payment method on their account will be physically charged for the specified amount. Of course, if wanted, you can use a hidden form field instead to specify the amount for the member, such as:

```
<input type="hidden" name="amount" value="14.95">
```

Last, you simply need to add a submit button to the form, such as:

```
<input type="submit" value="Deposit Funds">
```

3.5 Withdraw Funds

If wanted, you can also create your own HTML form, instead of using the *withdraw_funds.htmlt* template, allowing members to request a funds withdrawal from your account. First, you need a *<FORM>* tag, which will be:

```
<form action="index.cgi?~authtext~&action=withdraw_funds" method="POST">
```

The only required text field is to allow your members to specify the amount, they would like to withdraw, which will be:

```
<input type="text" name="amount">
```

Last, you simply need a submit button, such as:

```
<input type="submit" value="Withdraw Funds">
```

3.6 All Other Templates

There is no way to substitute a web page for the other HTML templates, except those described above. For all other templates, you must use the actually template located in the */xflow_data/htmlt/* directory. This is because the software must process, and / or retrieve certain database information, before displaying the template.

4.) Reverting to a Database Backup

There may be times when you need to revert your database to a backup. Assuming your crontab file has been successfully loaded, xFlow will automatically run every night at 1:00am server time, and depending on your settings, will backup your member database. This chapter explains exactly how to revert your database to a backup. If you do not feel comfortable doing this, customer support can revert your database for you.

1.) Login to your web server via Telnet / SSH.

2.) All backup files are located in the */cgi-bin/xflow/backup/* directory of your server. The files are named by the date the backup was taken. Pick the backup file you would like to use, and unpack it by typing:

```
tar –xzf FILENAME.tar.gz
```

3.) This will unpack the /xflow_data/ directory from the backup file. First, you need to either delete or rename the existing /xflow_data/ directory on your server, which is currently in use.

4.) Move the recently unpacked /xflow_data/ directory, which should be on your server at */cgi-bin/wsr/backup/xflow_data/* to the correct location. Move it to where the /xflow_data/ directory you just deleted or renamed used to be located. You can easily move the entire directory, by typing something such as the following at the SSH prompt:

```
mv wsr_data /home/username/xflow_data
```

5.) Change to the */xflow_data/tools/* directory on your server, and run the general.pl script by typing:

```
./general.pl
```

6.) Drop all current database tables by typing 'DROPDB' at the prompt, without the quotation marks.

7.) Exit the general.pl script by typing "0" at the prompt

8.) There will be a SQL dump file located on your server at */xflow_data/dump.sql*. Change to the */xflow_data/* directory, and load the dump file into your database by typing something such as:

```
mysql –uDBUSER –pDBPASS DBNAME < dump.sql
```

That's it! If you've successfully completed all above steps, you're entire member database will have been successfully reverted to the backup. You can now continue running your program.

5.) Transfer Member Database

This section explains how to transfer your member database from one web server to another. You may need to do this, for example, if you're transferring to another web host. If you are unable to complete the transfer yourself, customer support will for an additional fee. To transfer your database to another server, complete the following steps:

1.) Do a clean install of xFlow on the new server. Do not run First Time Setup on the new server.

2.) Login to the old server via Telnet / SSH

3.) Change to the */xflow_data/tools/* directory of your old server, and run the general.pl script, by typing the following at the SSH prompt:

```
./general.pl
```

4.) Backup your member database by typing 'BACKUP' without the quotation marks at the general.pl prompt. A success message will be displayed, giving the exact filename of the backup file. Exit the general.pl script by typing '0' at the prompt.

5.) Transfer the backup file to the new server, into the same directory as the */xflow_data/* directory is located. When you view the contents of the directory, you should see both, the backup file and the */xflow_data/* directory.

6.) Login to the new server via SSH, and rename the current /xflow_data/ directory, by typing the following at the SSH prompt:

```
mv xflow_data xflow_data_old
```

7.) Unpack the backup file you transferred over from the old server, by typing the following at the SSH prompt:

```
tar –xzf FILENAME.tar.gz
```

8.) Open the */xflow_data_old/xflow.conf* file in a text editor, and copy lines 3 – 14 *(@cmds - $in{'gif_url'}).*

9.) Open up the */xflow_data/wsr.conf* file, and replace the lines you previously copied as needed. Delete lines 3 – 14 in the xflow.conf file, and paste in the copied lines.

10.) Delete the */xflow_data_old/* directory from your server, by typing the following at the SSH prompt:

```
rm –rf xflow_data_old
```

11.) Open up your web browser, and view the dbinfo.cgi script, which should be located at a URL such as:

http://www.domain.com/cgi-bin/xflow/dbinfo.cgi

12.) Enter the database information for the new server, and submit the form. The next screen should ask you to install the database. Do not do this, and instead, close the web browser.

13.) There will be a SQL dump file located on your server at */xflow_data/dump.sql*. Change to the */xflow_data/* directory, and load the dump file into your database by typing something such as:

```
mysql –uDBUSER –pDBPASS DBNAME < dump.sql
```

That's it! If you have successfully completed all above steps, your member database will have been successfully transferred from one web server, to another. You can now continue running your business as normal.