

.com Solutions Inc.



FmPro Migrator - PHP to LiveCode Conversion

FmPro Migrator - PHP to LiveCode Conversion

1 PHP to LiveCode Conversion

1.1	Introduction - PHP to LiveCode Conversion	4
1.2	Step 1 - Create FmPro Migrator Project File	6
1.3	Step 2 - Select Conversion Options & Convert PHP Files	10

PHP to LiveCode Conversion

Introduction - PHP to LiveCode Conversion

This document provides an explanation of the steps required to convert PHP scripts to LiveCode scripts using FmPro Migrator Platinum Edition.

Revision 02

9/7/2013

[Updated LiveCode icon, screenshots, and changed revTalk to LiveCode.]

About the PHP to revTalk Conversion Process

The PHP to LiveCode conversion process is designed to convert all of the PHP or HTML files within a source directory, including all subdirectories. Most web servers are configured to process files having the .php file extension and send the output to a web browser. However some web servers are configured to execute php code within `<?php ?>` tags inside of files having .htm or .html file extensions. FmPro Migrator is designed to read both .php and .htm/html files during the conversion process.

Each file within the source directory is read into memory and analyzed on a line by line basis. Keywords and operators are read and converted to the equivalent keywords and operators in LiveCode. Variables are renamed to remove the leading \$ symbol and semicolons are removed from the end of each line of code.

PHP Code Processing Requirements

The following situations will require manual processing:

- 1) Poorly formed PHP code will not get fully processed. For instance, keywords, operators and variables should generally be separated from each other by at least one space character.
- 2) One instruction per line can be parsed correctly. The additional instructions of PHP code on the same line won't be completely converted.
- 3) PHP Server variables which are not available in revServer (i.e. `$_SERVER[PHP_SELF]`).
- 4) PHP internal library functions (i.e. `number_format()`).
- 5) PHP external libraries (i.e. `DB::connect()`, `query()`, `fetchRow()`).
- 6) Syntax errors or other types of problems which prevent the PHP code from executing won't be corrected.
- 7) PHP object oriented code features won't be converted, but will remain in the converted code for reference purposes.
- 8) Curly braces `{}` used for variable interpolation will be passed thru unconverted. These curly braces should be removed manually.
- 9) Multiple variable assignments within the same instruction. These statements need to be separated into two separate instructions.
- 10) Creation or addition of PHP arrays using the `array()` function. These statements need to be manually put into separate statements in LiveCode.

11) PHP associative array references using a text value containing spaces won't be re-mapped into LiveCode variables.

PHP Omitted Functions and Keywords

Some PHP functions are purposely omitted from the conversion processing because they require manual work:

Some of these functions/keywords require manual processing because the PHP parameters are specified differently than the equivalent LiveCode functions. Other functions/keywords require a wider view of the overall flow of the code than can be achieved via a line by line conversion technique (i.e. replacing nested IF/ELSEIF statements with SWITCH/CASE statements).

1) `substr($variable1, 0,30)` - This function should be manually converted to something like `char 1 to char 31 of variable1` [LiveCode indexing starts at 1 not 0].

2) `str_replace()` - Use the LiveCode `replace` keyword:
`replace tContainer with "text to replace"`

3) `elseif` - Most `ELSEIF` code will be more easily readable if it is manually replaced with the LiveCode `SWITCH/CASE` statements.

4) `for` - There are enough differences in the PHP `FOR` statement parameters to require manually rewriting it with the LiveCode `repeat/for` statement:

`repeat for variable = 1 to/down to 20 step increment`

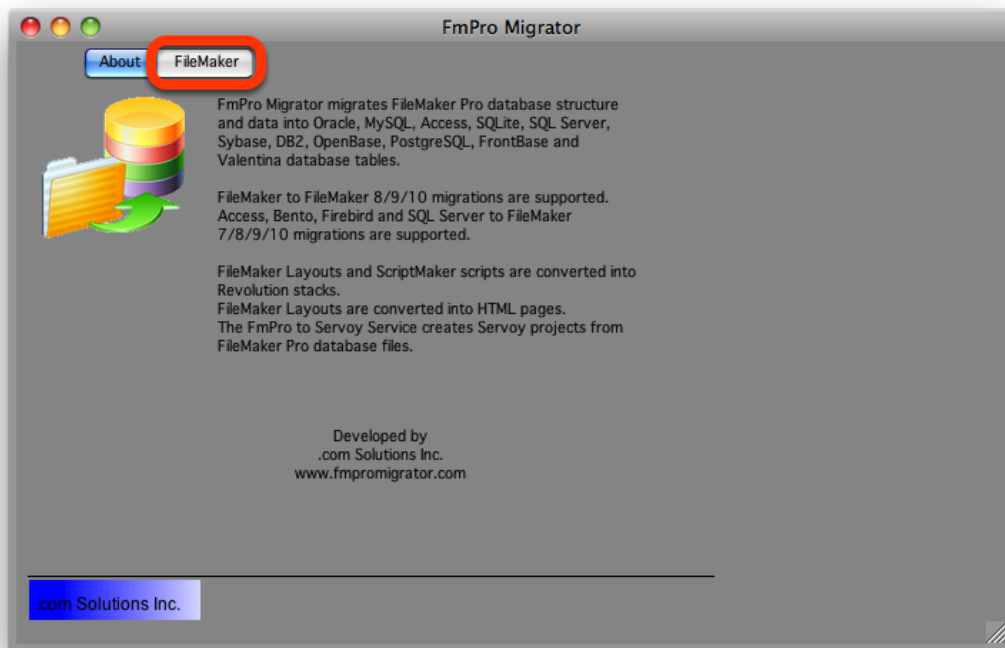
5) `foreach` - Rewrite manually using the `repeat for each key/element` and replacing the PHP value variable with a reference to the current data element.

6) `array_key_exists()`, `in_array()`, `array_search()` - Rewrite manually with references to the array and key value which is being searched. This can easily be done with the associated array feature in LiveCode, in which array keys can be either numeric or text strings. Reference the array using the requested key value, and check to see if the result is a non-empty result.

Step 1 - Create FmPro Migrator Project File

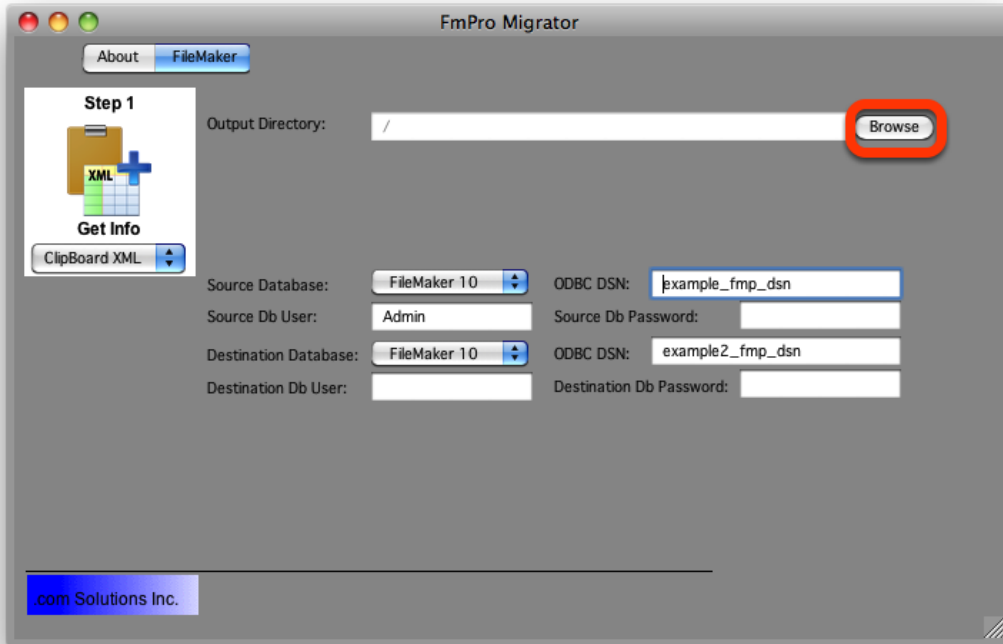
In order to perform a migration project, FmPro Migrator needs to create a MigrationProcess.db3 project file to store information about the migration project. Code conversion projects work a little differently than database conversion projects, so the Create Project File... menu is used to get the process started.

Open FmPro Migrator



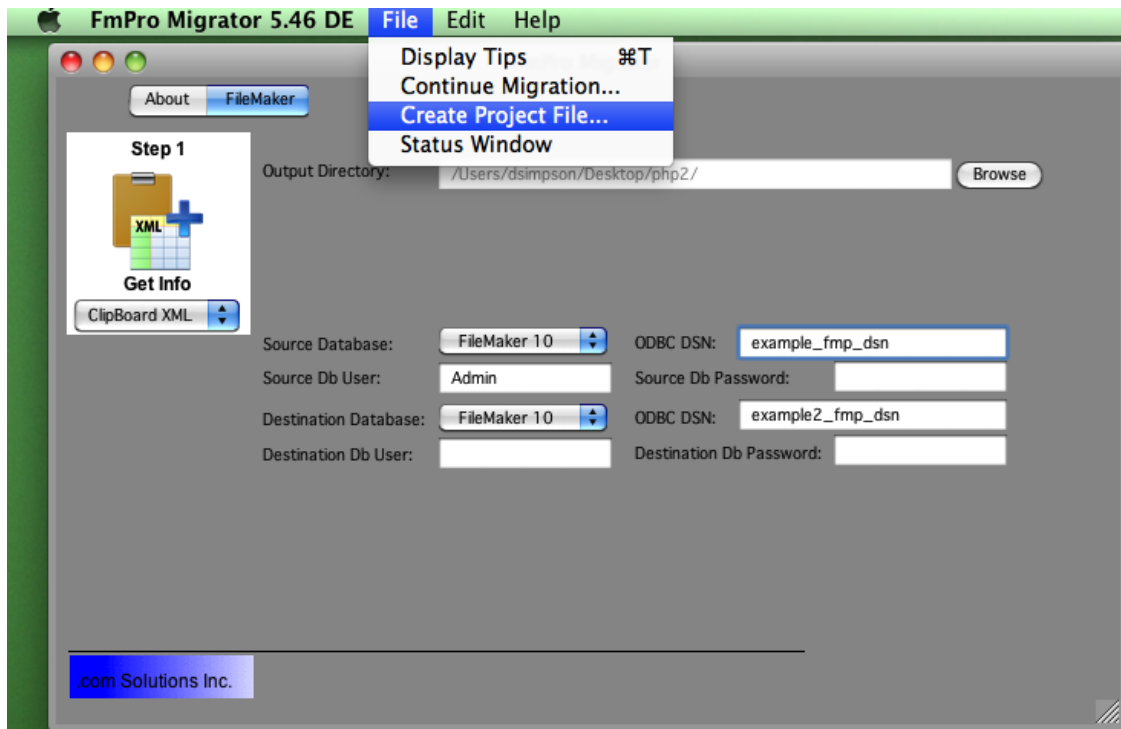
Click the FileMaker tab to select an output directory.

Click FileMaker Tab



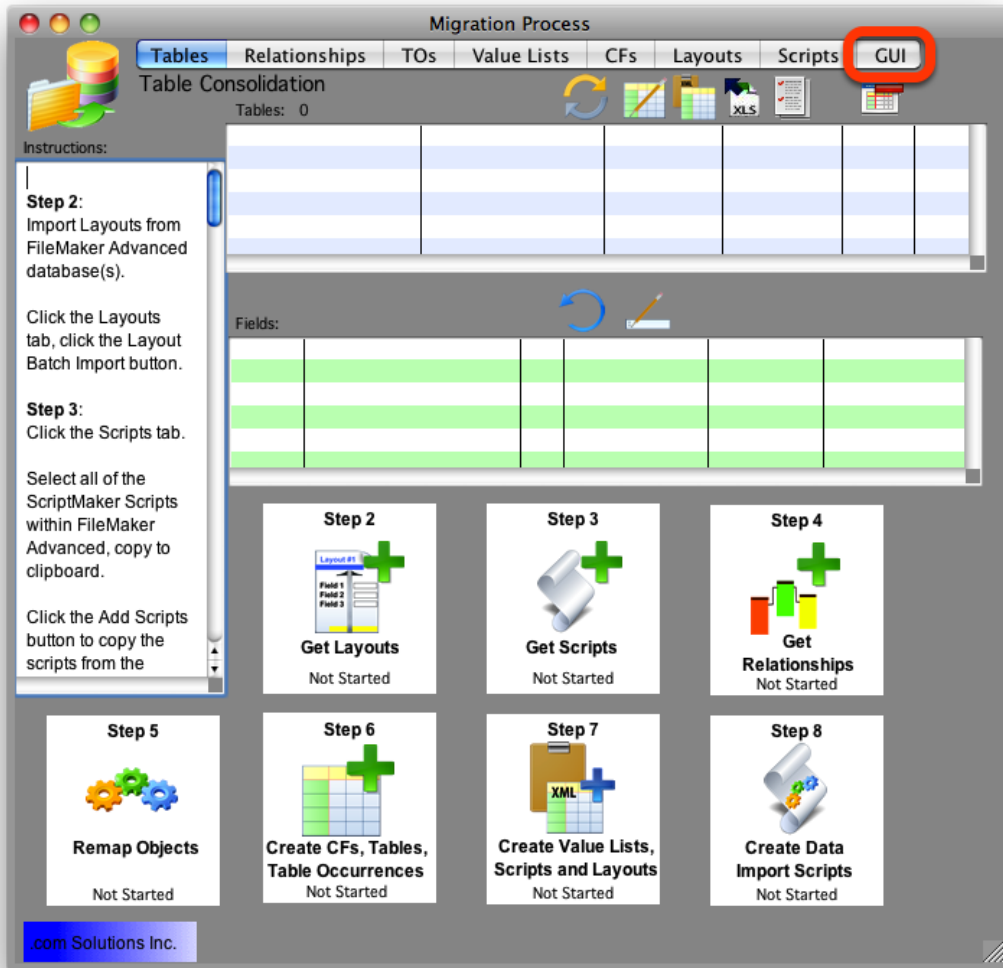
Click the Browse button to select the directory which will be used to store the FmPro Migrator project file. This directory can be the same output directory used for generating the converted scripts or stack file or it can be a different directory.

Select Create Project File... Menu



Select the Create Project File... item from the File menu. As soon as the FmPro Migrator project file has been created, the Migration Process window will open.

Click GUI Tab of Migration Process Window



Since a database migration is not being done, ignore the contents of the various database migration features, and click on the GUI tab button.

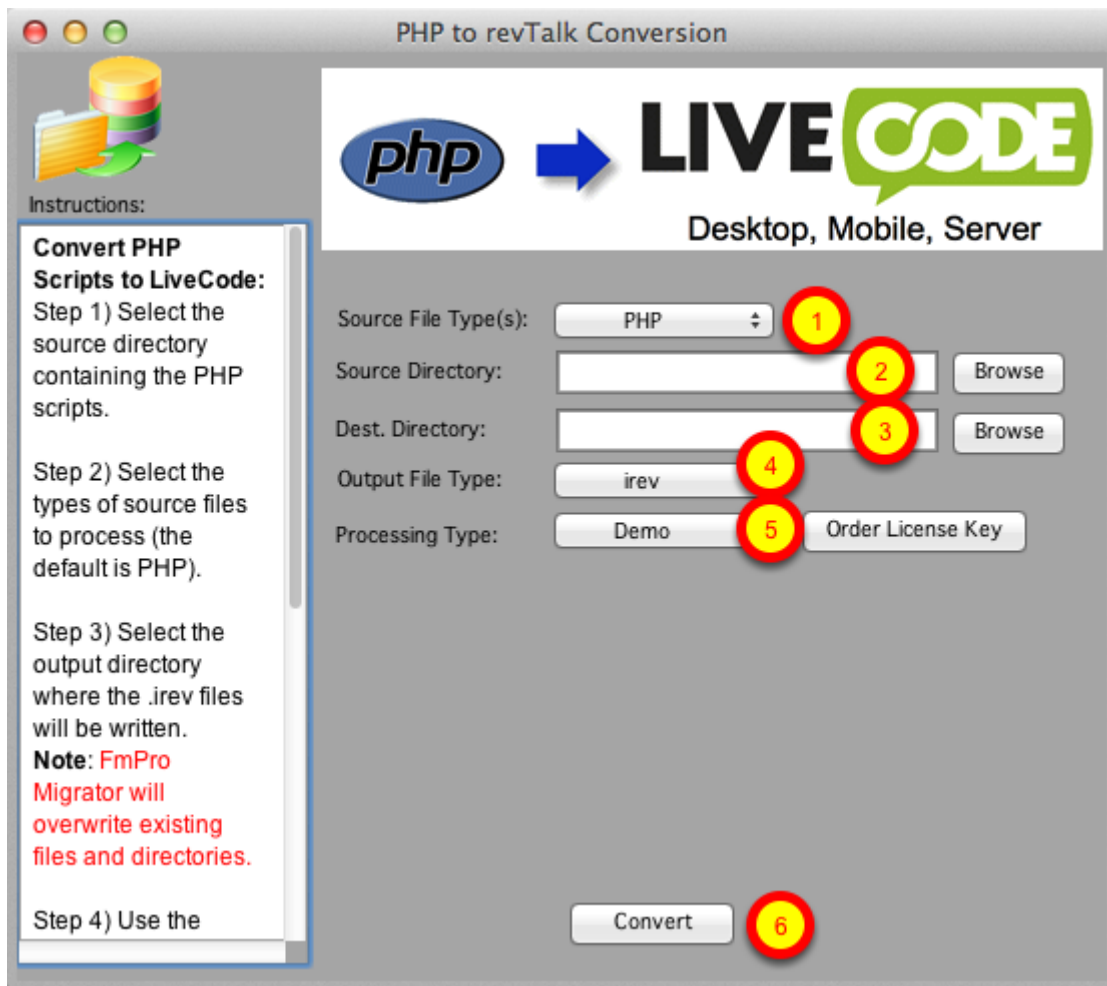
Step 2 - Select Conversion Options & Convert PHP Files

Click PHP to LiveCode Button



Click the PHP to LiveCode button to open the PHP to LiveCode window.

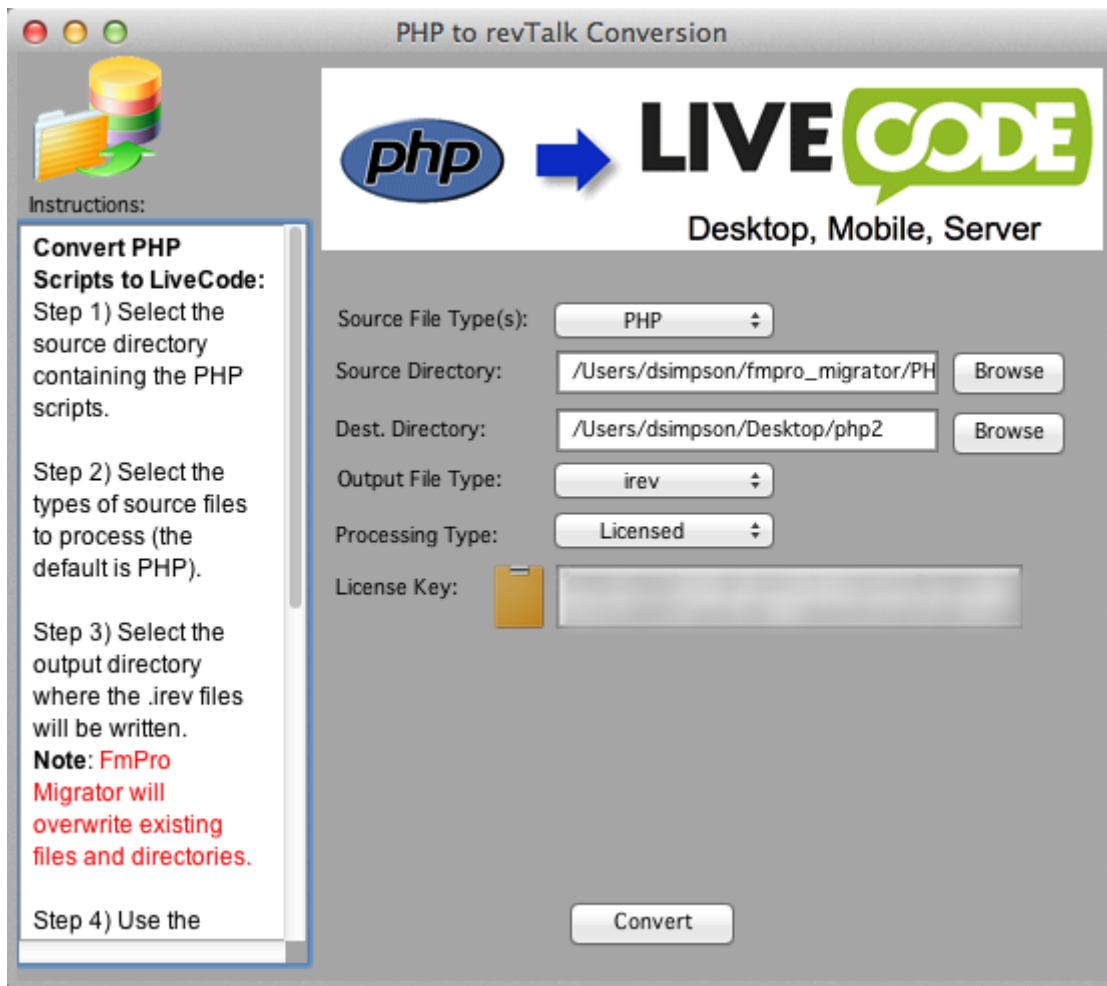
PHP to LiveCode Options



There are several options which need to be set prior to performing a code conversion project:

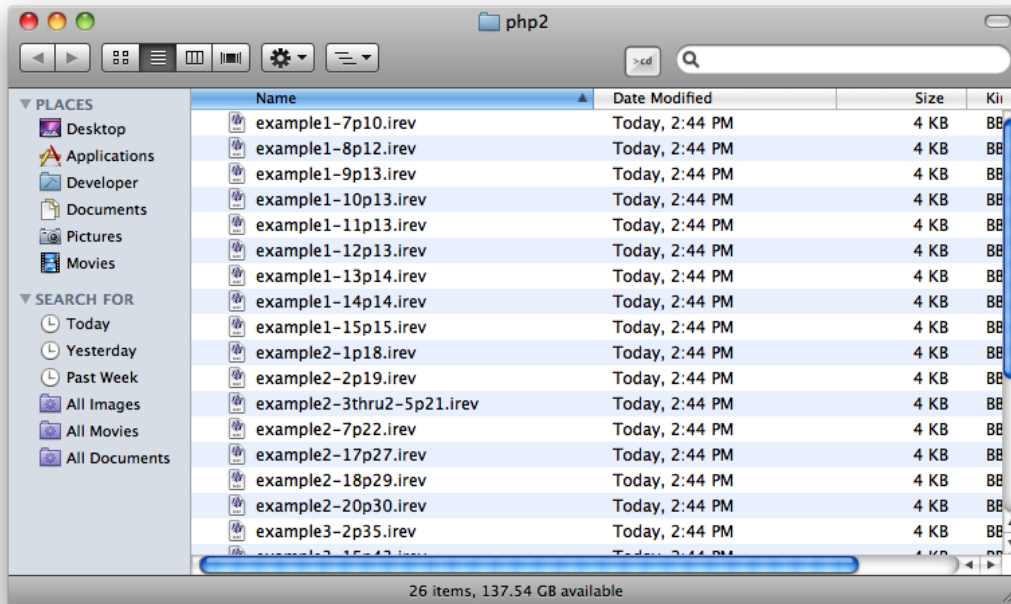
- 1) Source File Type(s): PHP or PHP,HTML - If you want to convert php code embedded within .htm or .html files, select the PHP,HTML option. (This is rare, most of the time you will just process files having the .php extension.) Otherwise, just use the default PHP option.
- 2) Source Directory - This is the top-level directory containing your PHP scripts. All enclosed directories will also be traversed and files within those directories will also be processed.
- 3) Destination Directory - This is the output directory where the converted files will be written.
- 4) Output File Type - The PHP files can be converted into .irev files or a single Rev stack having a card representing each converted source file.
- 5) By default, the PHP to LiveCode process operates in Demo mode. In Demo mode, 5 files of unlimited length will be processed. Ordering a license key removes this limitation.
- 6) Click the Convert button to convert the PHP files.

Conversion Results



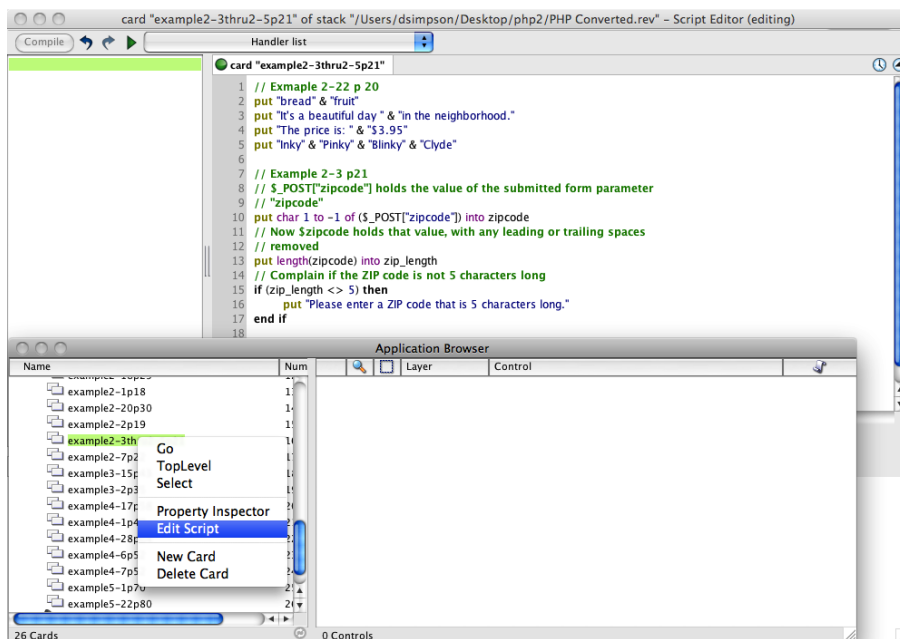
After clicking the Convert button, FmPro Migrator converts each of the files and displays the conversion results.

.irev Converted Files



The generated .irev files shown in the output directory. When generating .irev files, the <?php tags are converted into <?rev tags.

PHP Converted.rev Stack with Converted Card Scripts



If the stack output file type is selected, a stack named PHP Converted.rev will be created within the output directory. Each PHP script is converted into a card having a name consisting of the subdirectory name and script name. Select any card in the Rev Application Browser and

right-click on the card to select the Edit Script contextual menu item. The script will be opened in the Rev editor.