

Creating a data a basic transformation step by step with data load to a target data base.

Before trying this process, you must read and practice the document “Creating a data transformation step by step”, so, I assume that you know how to get to the EplSite ETL control panel and you have created the data base connection EplSiteETL_SQLite2 and the ETL Scheme:Step By Step Sample because, we are going to use them in this exercise.

An important note here is that at this point you must not worry about the catalog tables, specially if you are not going to use them but EplSite ETL needs the parameter to be set when the Scheme was created.

Inside EplSite ETL control panel, scroll down until **Data Transformation Definitions Maintenance.** And create the new transformation process:

Enter the main data in the form:

Data Transformation Definitions Maintenance.

Scheme Code:	StepByStep - Step By Step Sample ▼
Transformation Code:	6666
Description:	Get EplSite Modules And Insert To General Data Table
Show In Menu:	<input checked="" type="radio"/> Yes <input type="radio"/> No
Perl Script Name And Extension To Process This Transformation.:	ProcessLayoutDefinition.prc

Scroll down until section: **Perl Script Before Query**

This section is executed once and at the beginning of the process so, here we are going to delete the data from our table and we will prepare our insert query for faster insertion.

Perl Script Before Query, this section can be used to execute any preprocessing before getting data from query script below (This can be used for validation or deleting data before executing the main query.) There are 2 global variables:
\$globalp->{DBSourceConn}, \$globalp->{DBTargetConn}
\$globalp->{DBSourceConn} is where the data are read from.
\$globalp->{DBTargetConn} is where the data are going to be after transformation.:

```
1 my $DeleteQuery = "DELETE FROM general_data";
2 $DeleteQuery .= " WHERE kindofdata = 'EplSiteModules'";
3
4 if( defined($globalp->{DBTargetConn}) )
5 {
6     $globalp->{DBTargetConn}->do($DeleteQuery)
7     or die "Can not execute delete query:DeleteQuery";
8
9     my $InsertQuery = "INSERT INTO GENERAL_DATA (KINDOFDATA,DATAFIELD1,DATAFIELD2";
10    $InsertQuery .= ",DATAFIELD3) VALUES (";
11    $InsertQuery .= "?, ?, ?, ?)";
12    our $ModulesInsert = $globalp->{DBTargetConn}->prepare($InsertQuery);
13 }
14
```

Now scroll down until field: [Perl Script To Build The Query For The Layout](#), this is the query to get the data:

and enter the script to get the data:

Perl Script To Build The Query For The Layout, this is the query to get the data:

```
1 #This is a Perl script sample to filter addresses data, you must changed to fit your needs
2 $MyQuery = "SELECT title, custom_title,active FROM eplsite_modules";
3
4 # At this point you could call the function "get_data_from_main_query($MyQuery)"
5 #to get the data:
6 $QueryDataFound = &get_data_from_main_query($MyQuery);
7 #get_data_from_main_query returns 0 or 1 after processing
8 # 0=Not Data Found, 1= Data Found.
9
10
```

Now in the next field(**Script Before Process Transformation To The Data From Query**), leave as showed:

Script Before Process Transformation To The Data From Query above, this is in case you want do something with the data before processing the layout.
In order to process the layout you must set value to variable \$ContinueProcessingData to 0 or 1 (0=No Process,1=Process)
This variable is used as a condition to process the layout for actual record or continue with next record from query.

```
1 $ContinueProcessingData=1;
```

Scroll down until section **Script After Process Record Transformation**

Since this section is executed after each record has been transformed, we are going to do the data insertion here:

Script After Process Record Transformation

Here an additional transformation can be done or the insert to a table in a target database connection, the target database connection selected at the moment of executing the transformation can be used, it is stored in \$globalp->{DBTargetConn}:

```
1 if( defined($globalp->{DBTargetConn}) )
2 {
3     $ModulesInsert->execute($TransformedValues[1],$TransformedValues[2],$TransformedValues[3]
4     , $TransformedValues[4])
5     or die "Data can not be inserted in general_data table";
6 }
```

Lets go to the next section:**Script After Process Layout Transformation To The Data From Query**

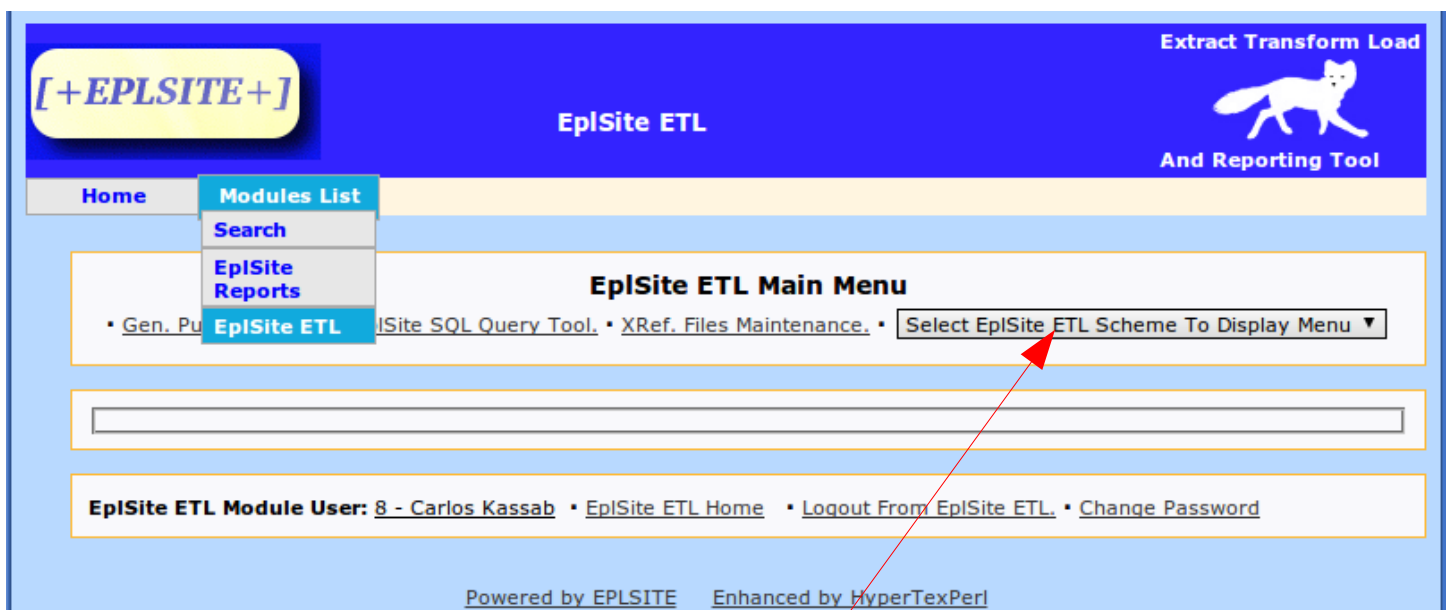
If this transformation process is going to be executed only once without any loops inside it, the insert query can be finished like this:

Script After Process Layout Transformation To The Data From Query
Here you could close your queries and disconnect any database.:


```
1 if( defined($globalp->{DBTargetConn}) )
2 {
3     eval{
4         $ModulesInsert->finish();
5     };
6     undef $ModulesInsert if($@);
7 }
```


Scroll down until the button <<Create Data Transformation Definition>> and click on it.

Now go to EplSite ETL menu, we need to set where each field will go in output:



From the main menu, select your new etl scheme from the combo box and you will see the screen below:



Extract Transform Load

 And Reporting Tool

EplSite ETL

[Home](#) [Modules List](#)

EplSite ETL Main Menu

[Gen. Purpose Table.](#) • [EplSite SQL Query Tool.](#) • [XRef. Files Maintenance.](#) • [Select EplSite ETL Scheme To Display Menu ▼](#)

Menu For EplSite ETL Scheme: Step By Step Sample.

[Setup Export Layouts.](#)

Source DataBase Connection: [Select Source DataBase Connection ▼](#)

Target DataBase Connection: [Select Target DataBase Connection ▼](#)

Layout To Process: [Select Layout To Process ▼](#)

or • **Independent Perl Script:** [Select Script To Execute ▼](#)

Only for use when exporting to pipe delimited file db target connection.

☒ All Records

☐ Only Records With Correct XRef.

☐ Only Records With Catalog Entry Not Found.

☐ Only Records With XRef Error.

☐ Records With Catalog Entry And XRef Correct.


Include File Header?
☐ Numeric ☐ Descriptive ☒ Both ☐ No Header


Preview Query?
☐ Yes ☒ No

EplSite ETL Module User: [8 - Carlos Kassab](#) • [EplSite ETL Home](#) • [Logout From EplSite ETL.](#) • [Change Password](#)

Powered by EPLSITE Enhanced by HyperTexPerl

Now click on Setup Export Layouts and you will see the next screen:



Extract Transform Load

 And Reporting Tool

EplSite ETL

[Home](#) [Modules List](#)

Layouts Setup For Scheme:Step By Step Sample

• Fields terminated by: No file chosen

• [Maintain Data Transformation Scripts \(DTS \)](#)

• Layouts: [Select Layout ▼](#)

EplSite ETL Module User: [8 - Carlos Kassab](#) • [EplSite ETL Home](#) • [Logout From EplSite ETL.](#) • [Change Password](#)

Powered by EPLSITE Enhanced by HyperTexPerl

Now select your new layout from the combo box, click on maintain and create your layout:

Layouts Setup For Scheme:Step By Step Sample • Fields terminated by: No file chosen

• [Maintain Data Transformation Scripts \(DTS \)](#) • Layouts:

Layout Code: 6666 - Layout Description:Get EplSite Modules And Insert To General Data Table

Seqn.	Field Description	Catalog And Flds Sq. For Validation	Value Type	Value Assigned	Export Field
<input type="checkbox"/> 1	EplSite Modules	----- ▾	1 0 Constant Value ▾	EplSiteModules	Yes ▾
<input type="checkbox"/> 2	Title	----- ▾	2 0 Query Field ▾	title	Yes ▾
<input type="checkbox"/> 3	Custom Title	----- ▾	3 0 Query Field ▾	custom_title	Yes ▾
<input type="checkbox"/> 4	Active	----- ▾	4 0 Query Field ▾	active	Yes ▾
5		----- ▾	5 0 ----- ▾		No ▾

Note, you can see that we are using Query Field, this is the name of the query variable or the alias if you are using “as” in your query.

Now just click Save and that's it, we are done with our first data transformation.

Go back to the main menu of your scheme and reload it selecting it again from the combo box to run the data transformation recently created:

[+EPLSITE+] **EplSite ETL** Extract Transform Load  And Reporting Tool

[Home](#) [Modules List](#)

EplSite ETL Main Menu

• [Gen. Purpose Table.](#) • [EplSite SQL Query Tool.](#) • [XRef. Files Maintenance.](#) •

Menu For EplSite ETL Scheme: Step By Step Sample.

• [Setup Export Layouts.](#)

• **Source DataBase Connection:** • **Target DataBase Connection:**

• **Layout To Process:** or • **Independent Perl Script:**

Only for use when exporting to pipe delimited file db target connection.

☒ All Records ☐ Only Records With Correct XRef. ☐ Only Records With Catalog Entry Not Found.
☐ Only Records With XRef Error. ☐ Records With Catalog Entry And XRef Correct.

Include File Header?
☐ Numeric ☐ Descriptive ☒ Both ☐ No Header

Preview Query?
☐ Yes ☒ No

See Log For Transformation Run Number:46

Now, select the next field:

Source database connection:EplSite_SQLite2

Target database connection:EplSite_SQLite2

Layout to process, select the layout code we just created(6666).

For the other form fields, just leave them as shown in the screen above.

Click Execute Button.

The transformation will be executed, you can check in the `general_data` table using EplSite Query Tool:

[illegible]

You also can see the transformation log if you click on the link showed below:

[+EPLSITE+]

EplSite ETL

Extract Transform Load
And Reporting Tool

[Home](#) [Modules List](#)

EplSite ETL Main Menu
• [Gen. Purpose Table.](#) • [EplSite SQL Query Tool.](#) • [XRef. Files Maintenance.](#) • [Select EplSite ETL Scheme To Display Menu ▼](#)

Menu For EplSite ETL Scheme: Step By Step Sample.
• [Setup Export Layouts.](#)
• **Source DataBase Connection:** [14 - EplSiteETL_SQLite2 ▼](#) • **Target DataBase Connection:** [14 - EplSiteETL_SQLite2 ▼](#)
• **Layout To Process:** [6666 - StepByStep - Get EplSite Modules And Insert To General Data Table ▼](#) or • **Independent Perl Script:** [Select Script To Execute ▼](#)
Only for use when exporting to pipe delimited file db target connection.
☒ All Records ☐ Only Records With Correct XRef. ☐ Only Records With Catalog Entry Not Found.
☐ Only Records With XRef Error. ☐ Records With Catalog Entry And XRef Correct.
Include File Header?
☐ Numeric ☐ Descriptive ☒ Both ☐ No Header
Preview Query?
☐ Yes ☒ No
[Execute](#)
[See Log For Transformation Run Number:46](#)

EplSite ETL - Google Chrome

localhost:3333/eplsite/index.prc?module=EplSiteETL&option=ShowRunNumberLog&runnumber=46

[Close Window.](#) [Reload](#)

RunNumber	DateTime	ETLUser	ETLSchemeCode	TransformationCode	LogMessage
46	20130217094003	eplsite	StepByStep	6666	Run Number Creation
46	20130217094003	eplsite	StepByStep	6666	Pre-transformation checking.
46	20130217094003	eplsite	StepByStep	6666	Connecting data sources.
46	20130217094003	eplsite	StepByStep	6666	Executing script before query.
46	20130217094003	eplsite	StepByStep	6666	Executing layout script.
46	20130217094003	eplsite	StepByStep	6666	Running script after transformation definiton.
46	20130217094003	eplsite	StepByStep	6666	Transformation Running End.

Powered by EPLSITE Enhanced by HyperTexPerl