Facebook Link

Extension page: http://ecommerce.aheadworks.com/magento-extensions/facebook-link.html
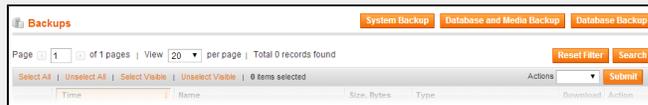
Installation

Installing Facebook Link extension

1. Backup your web directory and store database.
  ⌄ Click to view details
     You can make backup copies with any tool you find appropriate

     If you are going to use the native Magento backup function, navigate to System -> Tools -> Backups and perform System and Database backups
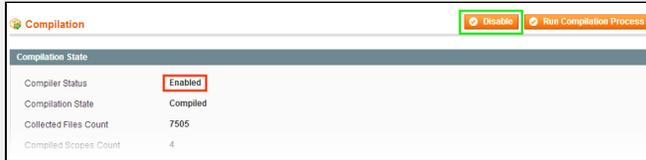

2. Log in to the Magento backend

3. Disable compilation
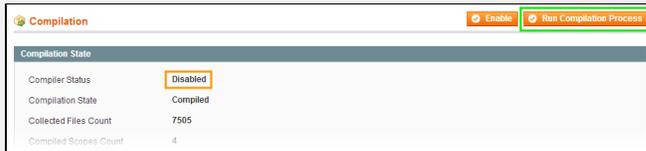
⌄ Click to view details

Navigate to System -> Tools -> Compilation.

If Compiler status is Disabled, you can skip to the next step

If Compiler is enabled, disable it.

IMPORTANT: after the extension is installed, you can enable the compilation again; IT IS CRUCIAL that you use "Run Compilation Process" function, not just "Enable button"

⊘ Installing an extension with the Compilation enabled will result in store downtime.

4. Flush store cache
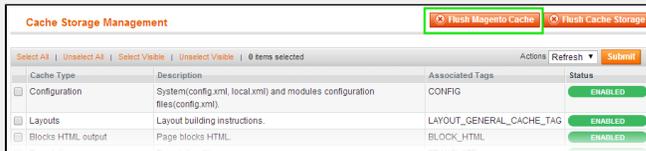
⌄ Click to view details

You can flush the store cache in 2 ways:

- via the backend:

Navigate to System -> Cache Management menu, and click Flush Magento Cache button

- via filesystem:

On your server, navigate to Magento root folder, then proceed to /var/cache/; delete all the content there.

5. Download the extension package from your account and extract the downloaded archive

6. Copy the content of /Step_1/ folder to your store's root directory

> ✅ If you are using a custom theme,
> ⌄ read the instructions here.
>> In case you are using a custom theme, it is recommended to copy the design files to your current theme's folders. In case there are several themes in use at the same store, the design files must be copied to each of them.
>>
>> - Find this folder in the package: /step_1/app/design/frontend/base/default/ ; copy its content to /app/design/frontend/[your_package]/[your_theme]/
>> - Find this folder in the package: /step_1/skin/frontend/base/default/ ; copy its content to /skin/frontend/[your_package]/[your_theme]/

7. Copy the content of /Step_2/ folder to your store's root directory

8. Flush store cache again; log out from the backend and log in again.

Registering application on Facebook

To register a Facebook app navigate to the Facebook Developers page. And follow the Facebook Register and Configure an App guide.

Configuration

All the extension configuration options can be found under System > Configuration > AheadWorks extensions.

As the extension introduces several aspects of the Facebook API functionality all settings are splint into five sections:

- Facebook Core;
- Facebook Connect;
- Facebook Like;
- Facebook Comments;
- Facebook Subscribe Button.

Facebook Core

In the Facebook Core settings section one is suggested to specify the Facebook App credentials
.

- Application ID - Facebook App ID number;
- Application Secret - Facebook App secret;
- Admin - this field defines a list of the Facebook user IDs to be considered as page administrators.

Application Settings		
Application ID		[STORE VIEW]
Application Secret		[STORE VIEW]
Admin		[STORE VIEW]
	▲ A comma-separated list of either the Facebook IDs of page administrators.	

Facebook Connect

Facebook Connect settings section allows setting up extension behavior related to logging in, automatic Facebook wall updates etc.

General options

- Enable Extension - allows enabling/disabling the extension;
- Title - allows defining the login button title.

General Options

Enable Extension	Yes	[STORE VIEW]
Title	Sign in with Facebook	[STORE VIEW]
	▲ The title of the button.	

Content Options

- Display on Login Page - defines if the Facebook login button should be displayed at the Login page;
- Display on Checkout Page - defines if the Facebook login button should be displayed at the Checkout page;
- Display Custom Buttons - allows disabling/enabling all custom buttons placed via CMS > Widget;
- Display Widget Buttons - allows disabling/enabling ALL Facebook Connect related buttons.

Content Options

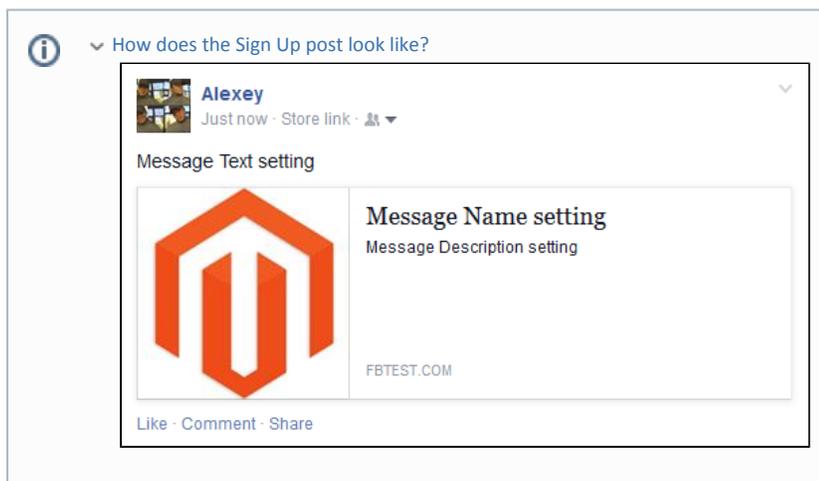|		
Display on Login Page	Yes	[STORE VIEW]
Display on Checkout Page	Yes	[STORE VIEW]
Display Custom Buttons	Yes	[STORE VIEW]
Display Widget Buttons	Yes	[STORE VIEW]

Wall Post

- Publish order on Facebook wall - defines if the order placed by the customer logged in via Facebook should be automatically placed on the customer's Facebook wall;
- Oder status to post order details to Facebook Wall - works in conjunction with Publish order on Facebook wall option and defines what status should obtain the order before it will be posted on the customer's wall;
- Customer Wall update - allows typing in a message that should be displayed on the customer's Facebook wall along with the order details;
- Action template - defines the look of the Facebook wall posts;
- Number of items to post - defines how many items from the order should be displayed on the customer's Facebook wall;
- Post product image - defines if the image of the product(s) purchased should also be displayed at the customer's wall;
- Use Wishlist Facebook Share Button - defines if the 'Share wishlist' button should be available at the customer's wishlist at the store;
- Customer Wishlist Share Wall update - defines a message for sharing a wishlist.

Wall Post

Publish order on FaceBook wall	Yes	[STORE VIEW]
Order status to post order details to Facebook Wall.	Pending	[STORE VIEW]
	▲ Indicate at what order status order details should be posted to user's Facebook wall.	
Customer Wall update	I've just purchased {items_count} item(s) at Mager	[STORE VIEW]
	▲ Publish a post into the stream on the user's Facebook Page's Wall and News Feed. Available variables: {items_count} - number of items purchased {store_link} - link to your store	
Action template	View {item_name} for {item_price} {item_link}	[STORE VIEW]
	▲ **Required field.** Available variables: {items_count} - number of items purchased {item_name} - item name {item_price} - item price {item_link} - link to the item purchased {store_link} - link to your store	
Number of items to post	0	[STORE VIEW]
	▲ 0 - all items	
Post product image	Yes	[STORE VIEW]
Use Wishlist Facebook Share Button	No	[STORE VIEW]
Customer Wishlist Share Wall update	I have some items in my wishlist at Magento Store {store_link} {wishlist_link}	[STORE VIEW]
	▲ Publish a wishlist into the stream on the user's Facebook Page's Wall. Available variables: {store_link} - link to your store {wishlist_link} - link to customer's Wishlist	

Facebook "Sign Up" Options

- Enable Facebook Wall "Sign Up" Message - allows defining a message to be posted at the customer's Facebook wall when first logging in to the store using Facebook account;
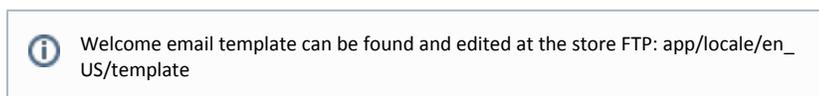
- Message Text - defines the message that will be displayed on the customer's Facebook wall upon registering at the store;
- Image URL - defines the image to be displayed along with the Message Text;
- Message Name - defines the store title to be displayed;
- Message Description - the description message to be displayed to the left of the store image defined with the Image URL.

Create New Account Options

- Enable Notifications - allows enabling/disabling welcome emails to customer;
- Welcome Email - defines a welcome email template;
- Email Sender - defines an email sender to be displayed in the From: field of the welcome email.

> Welcome email template can be found and edited at the store FTP: app/locale/en_US/template

Facebook Like

Facebook Like settings section allows setting up extension behavior related to Facebook like functionality.

General Options

- Button Plugin Code - defines the plugin code that should be used for the Like button functionality (leave the default value if you are not sure if you want to change it);
- URL to Like - allows defining the custom page that will be "liked" when customer clicks Like button disregarding of the Like button position at the store. Leave empty so the current page would be displayed as "liked";
- Send Button (XFBML Only) - defines if the Send button should be included along with

the Like button;
- Layout Style - defines the Like button's style:
  - standard;
  - button_count;
  - box_count;
- Show Faces - defines if the profile pictures of the customers who liked this page should be displayed below the Like button;
- Width - Like button block width in pixels;
- Height - Like button block height in pixels;
- Verb to display - defines the verb to display at the button:
  - like;
  - recommend;
- Color Scheme - defines the Like button's color scheme:
  - light;
  - dark;
- Title - defines the title of the Like post to display at Facebook (leave empty for default page title);
- Type - defines the type of the entity "liked" (e.g. page, article);
- URL - defines canonical, permanent URL of the page representing the entity;
- Image - allows defining images for the store entities "liked";
- Site Name - allows defining custom store name. Leave empty for default site name.

Content Options

- Display on CMS pages - defines if the Facebook Like button should be displayed at the CMS pages;
- Display on Home page - defines if the Facebook Like button should be displayed at the Home page of the store;
- Display on categories page - defines if the Facebook Like button should be displayed at the category pages;
- Display on products page - defines if the Facebook Like button should be displayed at the product pages;
- Display Custom Buttons - allows disabling/enabling all custom buttons placed via CMS > Widget;
- Display Widget Buttons - allows disabling/enabling ALL Facebook Like related buttons.

Facebook Comments

Facebook Comments settings section allows setting up extension behavior related to commenting particular store pages.

General Options

- Button Plugin Code - defines the plugin code that should be used for the Comment bu

tton functionality (leave the default value if you are not sure if you want to change it);
- URL to comment on - allows defining the custom page that will be "commented" when customer clicks Comment button disregarding of the Comment block position at the store. Leave empty so the current page would be displayed as "commented";
- Width - Comment block block width in pixels;
- Number of posts - defines number of the latest customer's comments to display in the Comments block;
- Color Scheme - defines the Comments block's color scheme:
  - Light;
  - Dark.

Content Options

- Display on CMS pages - defines if the Facebook Comment block should be displayed at the CMS pages;
- Display on Home page - defines if the Facebook Comment block should be displayed at the Home page of the store;
- Display on products page - defines if the Facebook Comment block should be displayed at the product pages;
- Display Custom Buttons - allows disabling/enabling all custom buttons placed via CMS > Widget;
- Display Widget Buttons - allows disabling/enabling ALL Facebook Comments related buttons.

Facebook Subscribe Button

Facebook Subscribe Button settings section allows setting up extension behavior related to subscribing with the Facebook account.

General Options

- Button Plugin Code - defines the plugin code that should be used for the Subscribe button functionality (leave the default value if you are not sure if you want to change it);
- URL to comment on - defines the Facebook page the customer will be subscribed to when clicking Subscribe button;
- Layout Style - defines the Subscribe button's style:
  - standard;
  - button_count;
  - box_count;
- Show Faces - defines if the profile pictures of the customers who subscribed the store Facebook page should be displayed below the Subscribe button;
- Color Scheme - defines the Subscribe button's color scheme:
  - light;
  - dark.
- Width - Subscribe button width in pixels.

Content Options

- Display on CMS pages - defines if the Facebook Subscribe button should be displayed at the CMS pages;
- Display on Home page - defines if the Facebook Subscribe button should be displayed at the Home page of the store;
- Display on products page - defines if the Facebook Subscribe button should be displayed at the product pages;
- Display Custom Buttons - allows disabling/enabling all custom buttons placed via CMS > Widget;
- Display Widget Buttons - allows disabling/enabling ALL Facebook Subscribe related buttons.

Customer Account Section

Facebook Link extension introduces new customer account section Facebook Connect.

The section allows customer decide if he wants the orders to be posted at his Facebook wall.

> ⚠ Post order details to Facebook Wall setting is enabled by default.

Custom positions

Facebook Link extension allows placing it's buttons and blocks using widget Layout Updates or CMS pages widgets.

The extension comes with the following widgets available:

- Facebook "Like" Button;
- Facebook Comments;
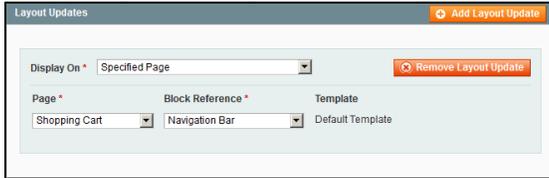- Facebook Connect;
- Facebook Subscribe.

Each widget represents related functional aspect of the extension.

Layout Update

Custom Facebook button widgets can be introduced to different store pages from CMS > Widget section.

1. Click Add New Widget Instance;
2. Specify required Facebook widget in the Type dropdown list;
3. Specify required Theme you want the widget to be displayed at;
4. Click Continue button;
5. Set required widget parameters in the Frontend Properties section;
6. In the Layout Updates section by clicking Add Layout Update button specify the store pages you want the widget to be displayed at.

Once the widget is saved required Facebook widget will be displayed at the set store pages.
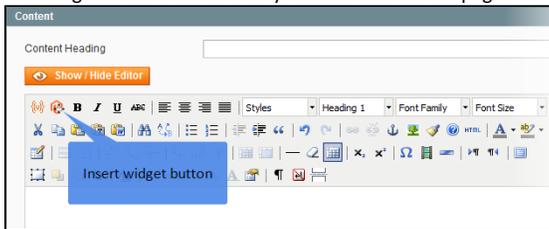

CMS Pages

Facebook Link widgets can be inserted in the CMS pages from CMS > Pages > Manage Content section.

1. Select existing or create new CMS page;
2. In the Content block click Insert Widget button of the WYSIWYG editor;
3. Select required Facebook Link widget.

The widget will be automatically inserted in the CMS page.

Uninstallation

1. Disable compilation, in case it is enabled.
2. Login to your FTP, navigate to app/etc/modules/
3. Open the files: AW_Facebook_Comments.xml, AW_Facebook_Connect.xml, AW_Facebook_Core.xml, AW_Facebook_Like.xml, AW_Facebook_Subscribe.xml and change the following line:

```
<active>true</active>
```

to

```
<active>false</active>
```

Now your Magento is unaware of the existence of the extension

4. Clear the cache under var/cache
5. Make sure that the site is working properly, otherwise roll back the changes and apply to our technical support.
6. If everything works fine, it is safe to delete the files of the extension.
7. In case you need to clean the database, make a backup and then run the following query in MySQL:

```
DROP TABLE 'mp_fb_like';
```

Note, if you are using the table with prefixes, you must specify them in all table names.
If you are not sure how to do that or expect any troubles with it, please contact your server administrator regarding the matter.
If you remove the tables as it is described above, you will need to configure the extension again after it is reinstalled.

Troubleshooting

After the extension installation the store gives an error, or blank page, or suggests to start Magento installation procedure.

Change the owner of the extracted extension files to the web server user and set 775 permissions on them. Clear the store cache and try again.

There is no aheadWorks extensions under my configuration section, or having the extension tab clicked I get a blank page, or Access Denied error.

Clear the store cache, browser cookies, logout and login again.

I've set up everything correctly, inserted the HTML code but there is nothing on that page.

Clear the store cache, clear your browser cache and domain cookies and refresh the page.

My configuration changes do not appear on the store.

Clear the store cache, clear your browser cache and domain cookies and refresh the page.