

Manual for MediaAttach v1.0

by Axel Guckelsberger
updated on 2008/03/09

Inhaltsverzeichnis

1. What is MediaAttach?.....	2
2. Installation.....	2
3. Update.....	2
4. Configuration.....	3
5. Quotas.....	4
6. Formats and Groups.....	4
7. Definitions.....	5
8. User Uploads.....	5
9. Admin Files / Imports.....	5
10. Content integration.....	6
11. Permissions.....	6
12. Display and Templates.....	7
13. Template Data.....	8
13.1. Upload formular.....	8
13.2. Upload array.....	8
13.3. Definition array.....	8
13.4. FileInfo array.....	9
14. Worth to mention.....	9
14.1. Embed external videos.....	9
14.2. Integration in the Profile module.....	9
14.3. Transform hooks.....	10
14.4. pnForum.....	10
14.5. Misc.....	10
15. Blocks.....	10
16. Module integration.....	11
16.1. Hooks.....	11
16.2. Display of files.....	11
17. Roadmap.....	12

1 What is MediaAttach?

MediaAttach is a hook module enhancing other modules with upload functionality. In the admin panel you can create different definitions describing the presets for each module. The goal behind MediaAttach is exculpating module developers from building own upload functionalities. MediaAttach is a Zikula only module.

2 Installation

1. copy MediaAttach into the modules directory of Zikula
2. regenerate the modules list within the Modules administration
3. initialise and activate MediaAttach
4. edit the hooks of your favored modules and activate MediaAttach for them
5. go to the admin area of MediaAttach
6. configure MediaAttach (see chapter 4)

3 Update

1. Remove all files of the old version (much files have become unnecessary)
2. copy MediaAttach into the modules directory of Zikula
3. regenerate the modules list within the Modules administration
4. initialise and activate MediaAttach
5. edit the hooks of your favored modules and activate MediaAttach for them
6. go to the admin area of MediaAttach
7. configure the new settings of MediaAttach (see chapter 4)

4 Configuration

In the configuration area several things must be set:

- **Upload Directory**

The directory in which all files and directories are being stored by MediaAttach.

The following points have to be considered here:

- The directory has to be specified absolutely. There must not be a slash at the end.
- The upload directory must be writable for the web server (chmod 777), therewith the files can be stored in it. You will see on the configuration page if this is true.
- In the optimum case the upload directory is outside the webroot directory which you can see one line above the input. This prohibits the directory (and the contained files) from being accessible over the internet. If there are open_basedir restrictions defined in PHP the directory must not be outside the directories allowed there. Further down you can see in the table with values from php.ini if open_basedir is set in your installation.
- If you can not create the directory outside the webroot for some reason (i.e. this is not allowed with shared hosting often), MediaAttach offers still a possibility to protect the uploads against direct access. Once the upload directory is writable you see a new formular under the configuration, with which you can create a .htaccess file at the push of a button.

- **Cache Directory**

The directory containing the image thumbnails MediaAttach creates. This directory must be inside the webroot, so that the thumbnails are readable by the user's browser. This directory must also be writable (chmod 777). It is being specified as a address being relative from the webroot.

- **Mail Function**

You can see this option only if you have the Mailer module from Zikula installed and activated. If you select this setting logged in users get the possibility to send mails fith files to themselves. You must fill out another field here to specify the maximum size a file may have to be allowed for sending it within a mail.

- **Activate quotas**

Turn the quota functionality on or off (more in chapter 5).

- **Allow handling of own files**

With this option you may allow your users to edit and delete their own files, independant from the normal permission system which will be explained further in chapter 11.

- **Default sizes of thumbnails**

MediaAttach creates thumbnails of uploaded images automatically. Here you can define their sizes. You can define multiple thumbnail sets to use alternatives for different purposes.

- **Shrink images automatically**

- **Maximum image size**

If you want MediaAttach shrinks all images exceeding a particular size.

- **Allow thumbnail cropping**

- **Behaviour of selection tool**

If this is set to not changeable, the default thumbnail size will be enforced in the selection tool. If this setting is changed to variable, users can also crop preview images in other sizes.

- **Activate frontpage in user area**

- **Categorization options**

MediaAttach does not have a public area as default. But you can unlock it and therefore use MediaAttach as an official download module. The files can be grouped by categories, modules and users. Please note that the public area contains all files for which the actual user has permissions – independent of what is set in the definitions of the according modules.

In the table with the values from php.ini you see important settings relating uploading of files. The following describes the meaning of each single directive:

- `open_basedir`
See above...
- `upload_tmp_dir`
Directory where files are stored temporarily after the upload
- `file_uploads`
Tells if file uploads are activated
- `upload_max_filesize`
Limits the maximum file size from server side
- `post_max_size`
Limits the maximum total size of a formular from server side
- `max_input_time`
Defines the maximum period the processing of an input may last

Version check

At the bottom you see another block with a version check. Here you are able to notice quickly and easily if new versions of MediaAttach exist. If this is true at this position the links to the download of the newest version will appear directly.

5 Quotas

If quotas have been enabled in the configuration, there exists a new link in the admin menu. Here you can define quotas for groups and users, whereat user quotas are overriding group quotas. If a user has reached a full quota he can no longer upload files.

6 Formats and Groups

The different file formats available in MediaAttach belong to several groups, at which one filetype can be in multiple groups.

Each group points to a virtual (!) directory. So you do not need to create this, but simply give it a name. Create new groups with individual mixtures of formats (as demonstrated in the group user defined example).

7 Definitions

Definitions allow modules using different upload formulars. You have to assign a definition to every module where you want to work with uploads. The explanation of the fields:

- **Module**
The module for which the definition is made for
- **Groups**
Defines the allowed file formats and therefore the allowed filetypes
- **Show uploaded files in the user area**
Tells if uploaded files are displayed above the upload formular. „*Only own*“ stands for the option that every user sees only those files which are uploaded by himself.
- **Send a mail during upload**
- **Recipient of the mail**
- **Maximum file size during upload**
Limits the size of files. Notice that this value must be smaller than the directives *upload_max_filesize* and *max_post_size* shown on the configuration page.
- **Download-Mode**
Defines how files are displayed. *Physical* stands for a simple link to the file, while *Inline* „tries“ to display the file embedded in the browser. More information regarding this you can find in chapter 12 being responsive to displaying uploaded files.
- **Naming convention**
Normally the original filenames are used after they have been cleaned from special chars etc. But you can also define here that files get a random name (md5 hash). The third option gives you the possibility to set a static name which is going to be combined with an unique number. So you could name your files in pagesetter *document<number>.extension* and in pnForum *forum<number>.extension* for example.
- **Number of files**
This is the number of files which can be uploaded at the same time. The default value is 1, the maximum I set to 8.

8 User Uploads

In this section you have an overview over the uploaded files. You can edit descriptions and titles or delete unwanted uploads. Furthermore the displayed files are sortable by different fields.

9 Admin Files / Imports

During installation MediaAttach will be activated for MediaAttach automatically. You can create a definition for it like for any other module. This allows you separate uploads within the admin section.

This area also contains a flexible and comfortable import of files from server directories. Just try it out!

MediaAttach does also support importing files and categories from other modules there. At the moment this import is available for Downloads 2, mediashare, PhotoGallery and pnUpper. Support for PNphpBB2 attachment mod and pagesetter is planned.

10 Content integration

MediaAttach works together with the wysiwyg editor xinha in [scribite](#). With this one can include MediaAttach files into articles or other content in a comfortable way.

The pndocs directory contains a [Guppy](#) plugin which can be used in Pagesetter. MediaAttach does also support the [Content](#) module, so that specific files can be integrated there without problems.

Furthermore MediaAttach also supports the needle functionality of the [MultiHook](#), so that an integration of uploads is still easily possible even without special plugins. If one inserts the expression *MEDIAATTACH{P-fileid|I-fileid}* into desired contents, it will be replaced by corresponding displays by the MultiHook. For example *MEDIAATTACHP-5* shows the file with ID 5 with the physical template, while *MEDIAATTACHI-5* uses the inline template.

11 Permissions

The permission schema of MediaAttach uses the component MediaAttach:: and three instance levels:

1. the particular module
2. the object id, i.e. at News *sid*, at pagesetter *tid_pid* or at pnForum *topic_id*.
3. the upload it, this is the ID of an upload file

A few example will make this clear quickly:

- pnForum:6: all files in topic 6
- pnForum:6:(3|5) only files 3 and 5 in topic 6
- News:(3|4|5): all files in articles 3, 4 and 5
- pagesetter:: all files in pagesetter
- pagesetter:4_3: all files in publication 3 of pubtype 4
- pagesetter:4_*: all files of pubtype 4
- News:[^34]: all files in all articles except 3 and 4
- pagesetter:2_[^34]: in all publications of pubtype 2 except 3 and 4
- pagesetter:2_\d*[^34]\d* all except those which ids contain 3 or 4

MediaAttach supports the following permission levels at the moment:

- OVERVIEW Viewing file lists and information
- READ Downloading and Sending files
- COMMENT Adding new uploads
- EDIT Editing files
- DELETE Deleting files
- ADMIN Complete admin section

Permission things are awkward for hook modules, since they can not recognize the module context they are included in. MediaAttach follows multiple approaches in parallel to offer an easy permission handling though.

- Independent from the Zikula permissions you can define within configuration, if users may edit and delete their own files.
- With pnForum MediaAttach performs an enhanced check and determines not only if a user wanting to reach a file has access to it, but even, if he has access to the category as well as the forum containing the according forum topic. In the roadmap (chapter 17) it is intended to realise this intensive checkup for all modules, indeed it is only integrated for pnForum for now.

12 Display and Templates

The display of uploaded files are depending on the download mode adjusted in the definition:

- *Physical* is a simple display whose entries basically consists of an image for the filetype, the download link as well as several information like filesize or download time for different bandwidths. This display we are calling *filelist*.
- *Inline* is an advanced alternative embedding files directly if possible. For instance this is possible for text and code fragments as well as some image formats, flash or audio/video files. Even the content of some archive formats (i.e. zip) can be processed by MediaAttach. This display we are calling *inlinelist*.

The most important templates of MediaAttach are:

- | | |
|--|-----------------------------|
| • MediaAttach_user_display.htm | Detail page for a file |
| • MediaAttach_user_delete.htm | Formular to delete a file |
| • MediaAttach_user_edit.htm | Formular to edit a file |
| • MediaAttach_user_filelist.htm | The file list for physical |
| • MediaAttach_user_filelist_single.htm | One entry in the filelist |
| • MediaAttach_user_inlinelist.htm | The file list for inline |
| • MediaAttach_user_inlinelist_single.htm | One entry in the inlinelist |
| • MediaAttach_user_uploadform.htm | The upload formular |

You can create an own copy for each module of each template. For that purpose simply append simply a `_<modulename>` to the template name. There are already templates for Pagesetter and pnForum included in this package which demonstrate this behaviour.

To customize the layout look into the file `pnstyle/style.css`. If you change the templates, please take care of keeping at least the defined id attributes, since otherwise some javascript functions could stop working probably.

There are two template sets for Pages serving as examples for the customizability of MediaAttach. The first one is an integration of the Highslide library. If you use only images in a particular module, you should try out this template – it's worth the few minutes. The second one integrates SWFUpload, a flash component, but this is not completely finished yet (compare chapter 17).

13 Template Data

Here is the important data described which is available in the most templates.

13.1 *Upload formular*

- authid auth key for the formular
- modname name of current module
- objectid id of current object
- definition current definition array
- allowadd right for uploading (true / false)
- redirect current url
- usequota quotas are being used (0 / 1)
- allowedquota allowed quota (in bytes)
- usedquota used quota (in bytes)

13.2 *Upload array*

- fileid id of the upload
- def* fields of definition array applied for the upload
- modname name of the module
- objectid id of the module object
- uid id of the user who uploaded the file
- username name of the user
- date datetime entry of the upload
- title title of the upload
- desc description of the upload
- extension file extension
- mimetype mime type
- filename filename
- filesize filesize
- dlcount number of downloads
- url the url where the upload has been performed
- format filetype image
- fileInfo FileInfo array

13.3 *Definition array*

- modname name of the module
- state definition is available (0 / 1)

- `did` id of the definition
- `displayfiles` file display (0 = none / 1 = all / 2 = own)
- `groups` list of group arrays
- `formats` list of format arrays
- `sendmails` send information mail at uploads (0 / 1)
- `recipient` mail recipient, if `sendmails` = 0
- `maxsize` maximum filesize in bytes
- `downloadmode` display mode (0 = physical / 1 = inline)
- `naming` file naming (0 / 1 / 2)
- `namingprefix` prefix, if `naming` = 2
- `anzfiles` maximum number of concurrent uploads

13.4 *FileInfo array*

- The available fields of this array you can see in the template `MediaAttach_upload_information.htm`.

14 Worth to mention

14.1 *Embed external videos*

MediaAttach can not only process uploads, but also include external videos from several providers like YouTube, Google Video or Dailymotion. If the group Flash or Media (and therewith the special file type `extvid`) is active, a new link appears to add external videos easily.

14.2 *Integration in the Profile module*

Users have the possibility to manage their own files within their own profile automatically. You have nothing special to do for this. But if you also want to include some information like the amount of uploaded files in the profile display of other users, please perform the following steps:

1. create the directory `system/Profile/pntemplates/config/`.
2. create a file called `usemodules` in this directory (without extension).
3. open this file with an editor and write `MediaAttach` into it. This allows the usage of MediaAttach plugins within Profile templates.
4. edit the template `system/Profile/pntemplates/profile_user_view.htm` to let it look like:


```
<!--[/foreach]-->
<!--[maprofileinfo uid=$userinfo.uid]-->
</tbody>
```

14.3 Transform hooks

The title and the description of uploads are transform hook aware. This means you can enable pn_bb smile or pn_bb click for MediaAttach for example.

14.4 pnForum

MediaAttach provides a deep integration into pnForum 2.7. You can find the necessary changes to pnForum in the file *pndocs/pnForum_Integration.txt*, until they are incorporated into pnForum officially. A demo of this integration can be found at <http://modulestudio.de/Forum/viewforum/forum/5> ;-)

14.5 Misc

MediaAttach should work fine with safe_mode activated. Feedback is appreciated. MediaAttach has been tested primary with pagesetter, pnForum, Pages and Reviews, but it should work with all modules being api compliant and supporting at least display hooks.

15 Blocks

At the moment MediaAttach has three blocks:

- **Newest files**
Shows the x newest files. If you want you can limit this to a selection of file formats (e.g. if you do not want videos to get listed there).
- **Specific files**
Shows one or more specifically selected files. This offers an easy possibility to provide the newest version of a tool or some important pdf documents for example.
- **Inline**
Shows x new or random files with inline template.

16 Module integration

16.1 Hooks

MediaAttach is not only a display hook for other modules, but it offers also some more hooks. In detail this contains a create, an update and a delete hook. Please note that the create and the update hook are both pointing to the function for creating new entries. The idea behind this is that you can upload new files during editing an article.

As an illustration of the might behind the Zikula's hook system this version of MediaAttach is working together with pnForum 2.7 in a special way. There you are able to upload files while writing posts directly. The delete hook takes care for removing relating files during the deletion of threads.

If you are a module author who also wants to include MediaAttach to your module it is completely sufficing if you call pnModCallHooks() whenever you create/update/delete an item to let the system gets the action to know. This is well illustrated in the example module.

16.2 Display of files

Important for displaying files and accordingly for linking to their downloads is that the output of a file must always happen with the function MediaAttach_user_download():

```
<!--[pnmodurlmodname="MediaAttach"func="download"fileid=$myfile.fileid]-->
```

The function MediaAttach_userapi_getupload() is used to get the upload array for a given fileid:

```
$myupload = pnModAPIFunc('MediaAttach', 'user', 'getupload',  
array('fileid'=>$myfileid);
```

To get multiple uploads at once, there is the function MediaAttach_userapi_getalluploads(), which is very flexible and knows much different parameters, for example filtering for a user and/or a module. You can find more information about this in the function header.

17 Roadmap

Version 1.1 („Stabilization“)

- complete SWFUpload templates
 - some open bugs
 - support title, description and categories
 - Upgrade to latest version which is now at <http://swfupload.org/>
- Fixes and improvements to top the current functionality off
- Function to assign files to module objects afterwards
- More migration to pnForm

Version 1.2 („Manifestation“)

- Support for other video providers (pnnextvidapi)
- Support for <http://pecl.php.net/package/uploadprogress>
- Blocks for manual output (per group directory, e.g. random image or random download)
- Improvement user frontend and categorie hierarchies

Version 2.0 („Expansion“)

- Migration to PNObject
- Management of different licenses
 - License form (e.g. CC type)
 - Creator
 - Content (e.g. Image text *CC Share-Alike Photo: Axel Guckelsberger*)
 - Temporary and restricted usage rights (e.g. online time)
- Uploads over multiple pages (temp store, i.e. during preview)

Version 3.0 („Perfection“)

- Generic enhanced permission check (see chapter 11)
- Improvement of inline display of document formats like OpenOffice, Word, Excel or pdf files
- Performance problem: many files in one directory
 - Disadvantage of subdirectories: does not work with safemode
 - let admin create sub folders or wait for PHP 6
 - combine with Categories