

EXT: Filelist

Extension Key: **dr_blob**

Version: **1.6.1**

Copyright 2000-2007, Daniel Regelein, <Daniel.Regelein@diehl-informatik.de>

This document is published under the Open Content License
available from <http://www.opencontent.org/opl.shtml>

The content of this document is related to TYPO3
- a GNU/GPL CMS/Framework available from www.typo3.com

Table of Contents

| | | | |
|--|----------|--|-----------|
| EXT: Filelist | 1 | Using the plugin..... | 14 |
| Introduction | 1 | Categories..... | 15 |
| What does it do?..... | 1 | Configuration | 17 |
| Screenshots..... | 2 | HTML Templates..... | 17 |
| Users manual | 4 | HTML Example Template..... | 17 |
| Creating a new record..... | 4 | Template-Marker Reference..... | 17 |
| Administration | 7 | TypoScript Reference..... | 19 |
| (System-) Requirements..... | 7 | Tutorial | 22 |
| Installation..... | 7 | Customizing [more] & [download]-links..... | 22 |
| Server settings..... | 9 | Having an own sysfolder for each language..... | 22 |
| Adding a new filelist to your Website..... | 9 | Known problems | 24 |
| Configuration of the plugin..... | 11 | To-Do list | 24 |
| Explaining the List-Types..... | 14 | Changelog | 24 |
| | | Credits | 25 |

Introduction

What does it do?

This Extension adds a frontend plugin to display a list of files (Binary Objects). The records are administrated in the backend.

This is actually no problem, but the extensions does a bit more than you can see: The files are not stored in the file system but in the database in a blob (BLOb = Binary Large Object) field.

Whenever a record of the type dr_blob is created it the uploaded file is quoted and stored in a database column.

You can store up to 2[^]32 byte per row; That are 4GB (!) in every row, depending on your disk space and the upload settings in `php.ini` and `my.cnf`.

Storing the files in the database has the following advantages:

1. It is very easy to restrict access to these records (f.e. with FE-Groups).
2. The records are a part of the backup of the database server.
3. The disk space of the application server isn't used that much.
4. The file does not exist physically on the hard disc...
5. ...so it is not possible to get the file via URL-Hacking.

Records can be categorized, and lists can be configured to show only a records having specific categories.

The list of the displayed records can be narrowed down by using the standard Typo3 features *starttime*, *endtime*, *hide* and

restrict access to a fe-group.

In the frontend you can choose between a *common list* of records, a customizable *list of special items* ("VIP-Items"), comparable to the LASTEST-view in `tt_news`, and a *personalized collection of items* each FE-user can define for himself. This mode allows authorized frontend users to subscribe several filelists on several pages to display a list of records of their interest. Therefore only the latest records are displayed. Using this function it is possible to generate personalized lists for every user.

The extension also has a search function to search for records in a special list.

The extension offers completely design freedom ensured by using the Typo3 inbuilt template engine.

Screenshots

• Common List of items in the FrontEnd

| Title | Author | Date | | |
|--------------------------------|---------|----------|--|-----------------------|
| Full Bundle | Jack | 23.03.06 | | [show] [download] |
| Only Component A | Adam | 25.03.06 | | [show] [download] |
| Only Component B | Simon | 01.03.06 | | [show] [download] |
| Component C+D | Jack | 01.04.06 | | [show] [download] |
| 30 days Evaluation Licence Key | Claudia | 09.03.06 | | [show] [download] |
| Readme | Adam | 28.02.06 | | [show] [download] |
| Installation Manual | Lisa | 24.03.06 | | [show] [download] |
| Installation Videomanual | Lisa | 24.03.06 | | [show] [download] |

• List of special (VIP) items in the FrontEnd

21.10.2005 09:37
Datenblatt Typo3
TYPO3 is a free Open Source content management system for enterprise purposes on the web and in intranets. It offers full flexibility and extendability...
[\[mehr\]](#)

05.10.2005 09:21
How to cook 40 humans
How to cook 40 humans
[\[mehr\]](#)

05.10.2005 08:58
Testdokument
Dies ist ein Testdokument
[\[mehr\]](#)

• List of items in the BackEnd

| Title | [Localization] | Localize to: |
|--------------------------------|----------------|--------------|
| Full Bundle | Default | |
| Only Component A | Default | |
| Only Component B | Default | |
| Komponente B | Deutsch | |
| Komponente C+D | Deutsch | |
| 30 days Evaluation Licence Key | Default | |
| 30 Tage Test Lizenzschlüssel | Deutsch | |
| Readme | Default | |
| Lies Mich | Deutsch | |
| Installation Manual | Default | |

- Input Mask for a single record

Hide:

Title: Full Bundle

Language: Default

Description: Paragraph: Default Text: Default

Normal **B** *I* U ~~X~~ ~~X'~~

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat.

Path: body

File categories

Selected: Items:

VIP-Record? Shall this record be shown in a vip-list?

Downloadcounter: This file has been downloaded 24 times.

Filename: FullBunde.zip

Filesize: 4.1 MB

Filetype: application/octet-stream

File:

Search records

Downloads suchen

book

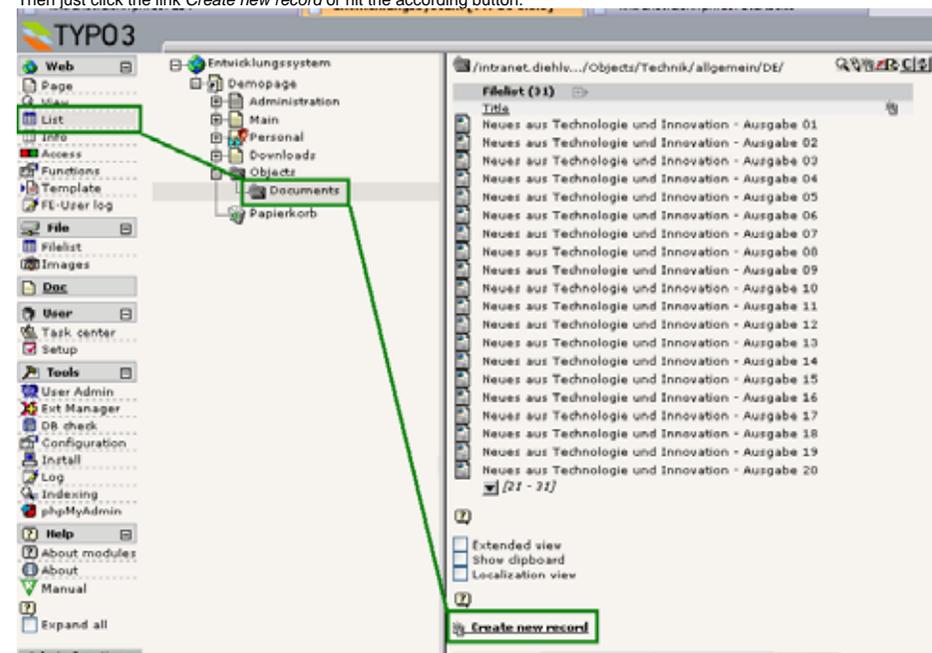
| Title | Date | |
|---------------------------------|------------|-----|
| eBook: How to cook 40 humans | 10/20/2005 | [M] |
| eBook: Codephrase "Hello World" | 10/27/2005 | [M] |

Users manual

Creating a new record

For creating new records you've to select the List-Mode. Select the sys folder containing the records from which the frontend plugin receives its data.

Then just click the link *Create new record* or hit the according button:



[Steps to create a new record]

Creating a file record

After choosing *Secure File* the following input mask will appear:

The screenshot shows the 'Creating a new file record' form in the TYPO3 Filelist extension. The form is organized into several sections:

- Hide:** A checkbox to control the visibility of the record.
- Start, Stop, Access:** Input fields and a dropdown menu for scheduling and access control.
- Title:** A text input field, marked with an exclamation mark (!) to indicate it is required.
- Language:** A dropdown menu with '[All]' selected.
- Description:** A rich text editor with a toolbar containing various formatting options like bold, italic, underline, and links.
- Path:** A text input field with 'body' entered.
- File categories:** Two list boxes labeled 'Selected:' and 'Items:' for categorizing the record.
- VIP-Record?:** A checkbox labeled 'Shall this record be shown in a vip-list?'.
- Downloadcounter:** A text input field showing '0' and a 'Reset Counter' button.
- Filename, Filesize, Filetype:** Text input fields for file metadata.
- File:** A text input field with a 'Durchsuchen...' (Browse...) button for selecting a file.

[Creating a new file record]

Fields marked with an Exclamation Mark (!) have to be filled out.

Fields of a File record

Title

This field contains the record's title and has to be filled out.

Language

Use this field to define the language of this record.

Original Translation

Use this field to attach the translated version to the default language record.

Description

This is the main text of a record. Depending on your editor component it is possible that the editor shown in the upstanding picture varies to yours.

Filename

The filename is only necessary if a file is attached. The field is filled out automatically when the record is saved for the first time. If the record is already saved you can set a custom filename here.

Filesize

The filesize field is set automatically when a file is attached to this record. It contains the number of bytes the attached file has.

You cannot edit this field.

Filetype

The filetype field is set automatically when a file is attached to this record. It contains the content type of the attached file.

You cannot edit this field.

File

This field is used to select the file that shall be attached to the document. To select the file from the hard disc hit the according button 'browse' and select the file.

When the record is saved, the fields *filename*, *filesize* and *filetype* will be set automatically.

VIP-Record

This checkbox prepares the record to be shown on the TOP-List of your website. If the hook is not set the record won't be listed in a top list.

Download Counter

This read only field show the number of times an attached file has been downloaded yet. Use the according button to reset it to zero.

File Categories

Use this field to categorize the record. The administrator can configure pages to show only records that have a specific category

Hide

This field is one of the four fields to control the visibility of a record. The record won't be shown if it is marked hidden.

Starttime

This field is one of the four fields to control the visibility of a record. The record won't be shown if the value of starttime is a point in the future.

Endtime

This field is one of the four fields to control the visibility of a record. The record won't be shown if the value of endtime is a point in the past.

Access

This field is one of the four fields to control the visibility of a record. The record won't be shown if current frontend user isn't a member of the group selected here. You can also configure this record to be only shown if a user is logged in (independent of the group membership) or if no user is logged in.

Creating a file category

After choosing File Category the following input mask will appear:

[Creating a new file category]

Fields of a file category

Hide

The category won't be shown on the website if it is marked hidden.

Title

This field contains the category's title and has to be filled out.

Administration

(System-) Requirements

The following table shows the versions of the different systems required for this Extension

| Typo3 | You should have at least Typo3 4.0.0. The extension may work with older Typo3 versions too, but there is no guarantee. |
|-----------------|--|
| PHP Interpreter | - Every version T3 works with - |
| mySQL | - Every mySQL Version the T3 versioning features work with - |

The Extension has not been tested with other database management Systems yet. If you do so feel free to inform me about the results.

Installation

You install this extension just like every other extension, using the extension manager.

Open the extension manager by clicking the according button and select the downloaded extension file.

Hit *Install Extension* to start the installation.

Extension Manager

Extension: **File list (dr_blob)** Information

Installing File list: DATABASE NEEDS TO BE UPDATED

Before the extension can be installed the database needs to be updated with new tables or fields. Please perform:

Add tables

- CREATE TABLE tx_drblob_content (
 - uid int(11) auto_increment,
 - pid int(11) default '0',
 - blob_type varchar(255) default ''
 - [...]
 - blob_type varchar(255) default ''
 - blob_data longblob,
 - PRIMARY KEY (uid),
 - KEY parent (pid),
 - KEY title (title)
 -);
- CREATE TABLE tx_drblob_personal (
 - uid_feusers int(11) unsigned default '0',
 - uid_pages int(11) unsigned default '0'
 -);
- CREATE TABLE tx_drblob_category (
 - uid int(11) unsigned auto_increment,
 - pid int(11) default '',
 - crdate int(10) default '',
 - tstamp int(10) unsigned default '',
 - cruser_id int(11) unsigned default '',
 - deleted tinyint(1) default '0',
 - hidden tinyint(1) default '0',
 - title varchar(255) default '',
 - PRIMARY KEY (uid)
 -);
- CREATE TABLE tx_drblob_category_mm (
 - uid_local int(11) default '0',
 - uid_foreign int(11) default '0',
 - tablenames varchar(30) default '',
 - sorting int(11) default '0',
 - KEY uid_local (uid_local),
 - KEY uid_foreign (uid_foreign)
 -);

Create upload folder

The extension requires the upload folder "uploads/tx_drblob/" to exist.
 Create directory "uploads/tx_drblob/":

"Accessing category records" [categoryStorage]
 Set the way you want to access File Categories. Select "StoragePID" for using the "General record storage page", and "IDList" to configure multiple Sysfolders using the Page-TS-config.
 StoragePID
 Default: StoragePID

"Enable Download-Counter Reset" [enableCounterReset]
 This option determines whether the Reset-Function for the Download-Counter is enabled- or not. Using this function can be restricted to special BE Groups through exclude fields. See the manual for more information.

 Default: 0

[performing the necessary Update Tasks]

Hit the button *Update* to perform the necessary update tasks. The Options „Enable Download-Counter Reset“ and “Accessing category records” are explained in the chapter below.

The plugin is now installed and may be selected in the plugin manager.

Configuration Options

These configuration option can be set using the Extension Manager or the Install tool.

Accessing category records

This option allows you to select where the plugin searches for category records. You can either select StoragePID to use the General record storage path of the Sysfolder (with is limited to one target), or you can select IDList to define multiple targets

using the PageTS of the sysfolder containing the file records.

Enable Download-Counter Reset

It determines whether it's possible to reset the field that counts how often a file has been downloaded yet. If this option is set to **true**, you'll have to enable it for non-admin-users using the exclude-fields.

Server settings

File upload limits

These settings define whether the php application server accepts file uploads and how big the files may be.

php.ini

| Param | Value |
|---------------------|---|
| file_uploads | (def. 'On') Take care that this option is enabled |
| post_max_size | Max. POST-size PHP will accept |
| upload_max_filesize | (def. 1M) Sets the maximum upload filesize. Please notice that raising <u>only</u> this value maybe won't have any affects; Also rise value <code>max_allowed_packet</code> in the my.cnf-file <i>Remember to restart the application server service (web server) after changing these value</i> |

```

; ;;;;;;;;;;;;;;;;;;;;;;;;;;
; File Uploads ;
; ;;;;;;;;;;;;;;;;;;;;;;;;;;

; Whether to allow HTTP file uploads.
file_uploads = On

; Temporary directory for HTTP uploaded files (will only work if
; enabled).
upload_tmp_dir = /tmp

; Maximum allowed size for uploaded files.
upload_max_filesize = 32M

```

my.cnf

| Param | Value |
|--------------------|--|
| max_allowed_packet | (def. 2M) This field defines the max. Packet size the mysql database will accept. Raise this value to increase the maximum filesize. <i>Remember to restart the database service after changing the value</i> |

Adding a new filelist to your Website

To insert a new Filelist-Element into a page perform the following steps:

1. Goto the page where the filelist shall be insert, and hit the button *Create Page Content*.
2. Select The Option *Pagecontent* --> *Click here for a wizard!*



3. Select the plugin *Filelist*. The number of elements listed here may vary from yours.

Table
A simple table with up to 8 columns.

Special elements

- Filelinks**
Makes a list of files for download.
- Multimedia**
Inserts a media element like a Flash animation, audio file or video clip.
- Sitemap**
Creates a sitemap of the website.
- Plain HTML**
With this element you can insert raw HTML code on the page.

Form elements

- Mail form**
A mail form allowing website users to submit responses.
- Search form**
Draws a search form and the searchresult if a search is performed.
- Login form**
Login/logout form used to password protect pages allowing only authorised website users and groups access.

Plugins

- General Plugin**
Select this element type to insert a plugin which cannot be found amongst the options below.
- News**
Versatile news system for TYPO3
- Filelist**
Adds a list of downloadable files that are stored in the database to the page

2: SELECT POSITION:
Now, click the position where you wish to place the element on the page:

LEFT NORMAL RIGHT BORDER

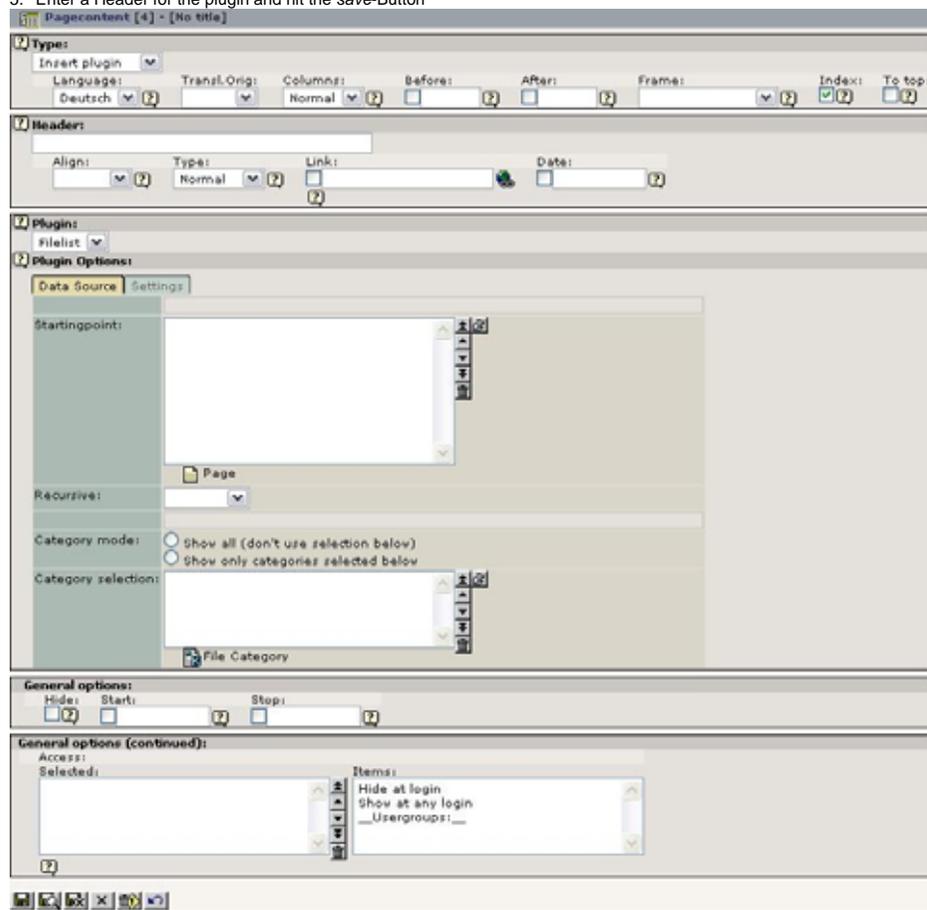
[Selecting the plugin using the Wizard]

4. Select the position where to place the plugin

2: SELECT POSITION:
Now, click the position where you wish to place the element on the page:

LEFT NORMAL RIGHT BORDER

5. Enter a Header for the plugin and hit the save-Button

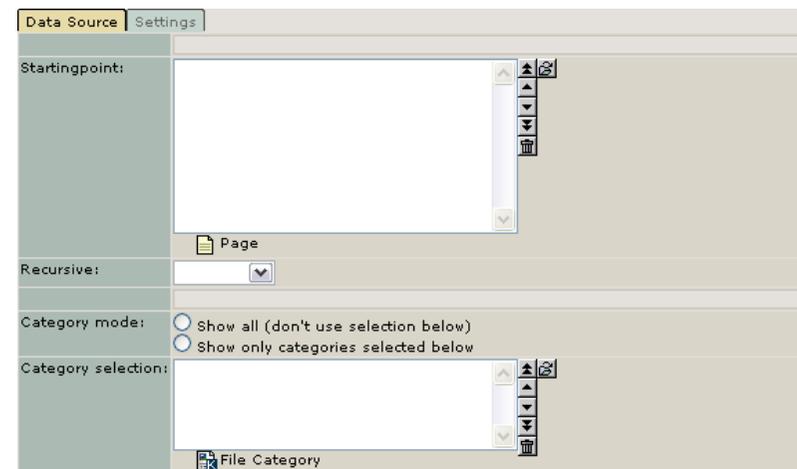


[Plugin Configuration]

6. To ensure the cleanness of your backend you should create a page from type sys folder as the data storage of the filelist entries.

Configuration of the plugin

Data Source



[Plugin Configuration Page 1]

Startingpoint / Recursive

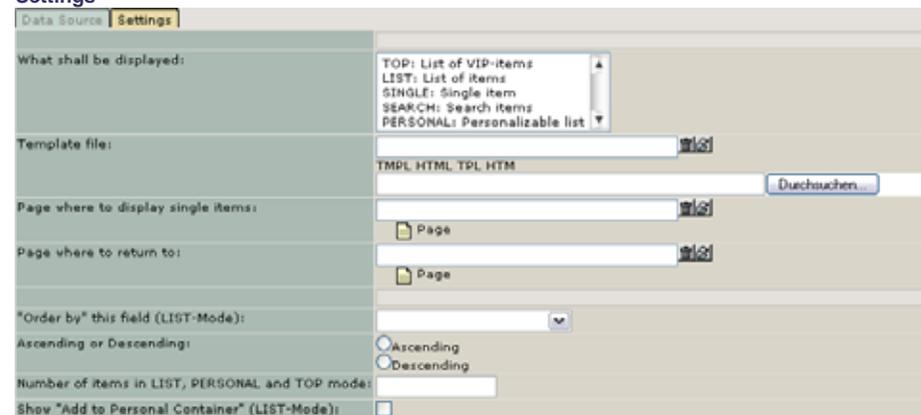
Select the sys folder(s) / pages where the records are being stored by selecting them in the field *Starting Point*.

The recursive level sets the level of sub folders that are also included.

Category mode / Category selection

Select whether to show all records, independent what categories they are associated with, or whether to show only records that have the categories selected in the field *Category Selection*.

Settings



[Plugin Configuration Page 2]

What shall be displayed

The field *What shall be displayed* allows you to choose of the four list-types the plugin offers. Each list-type has it's advantages. See the chapter **Explaining the List-Types** for detailed information about them.

The default value is the LIST-Mode.

Template File

Field to upload a template file for the plugin. An Example Template is placed in the sub folder `res` in the extension root directory. This value overwrites the template file set via TypoScript for the current plugin.

If no template is set, the extension uses its own standard templates.

Page where to display single items

Allows you to define a custom page on which the details of a record shall be displayed. That pages also needs to have a plugin of the type `dr_blob`.

If nothing is selected, the plugin sends the single record to the current page.

This is useful when you have other plugins / information on this page and don't want them to be affected by the filelist.

Page where to return to

You can select to which page the single mode shall return to. That pages also needs to have a plugin of the type `dr_blob`.

If nothing is selected, the plugin sends the single record to the current page.

This information only makes sense on pages with a `dr_blob`-plugin and the view-type `SINGLE`.

“Order by” this field (LIST MODE)

Set the default sorting of records in the LIST Mode (and so the list of results of the SEARCH mode too).

Have a look at the chapter Specific Sort Sequence for details on the *Custom Backend Sorting - Value*

Ascending or Descending

Set the default sorting direction of records in the LIST Mode (and so the list of results of the SEARCH mode too).

Number of items to list

Sets the number of items a list (no matter what type) contains.

Show "Add to Personal Container" (LIST-Mode)

If this value is set and the current plugin-mode is the list mode, authorized users will find a button that gives them the possibility to add the data sources of the current plugin to their personal list of item they are interested in. Once this button was clicked, it changes itself to remove the selection of this folder.

Field restrictions for editors

The following fields of a single record are exclude fields and allow access restriction.

| Fieldname | Function |
|------------------|--|
| Hide | This field is one of the four fields to control the visibility of a record. The record won't be shown if it is marked <i>hidden</i> . |
| Start | This field is one of the four fields to control the visibility of a record. The record won't be shown if the value of <i>start time</i> is a point in the future. |
| Stop | This field is one of the four fields to control the visibility of a record. The record won't be shown if the value of <i>end time</i> is a point in the past. |
| Access | This field is one of the four fields to control the visibility of a record. The record won't be shown if current <i>frontend user</i> isn't a member of the <i>group</i> selected here. You can also configure this record to be only shown if a user is logged in (independent of the group membership) or if no user is logged in. |
| VIP-Record | This check box prepares the record to be shown on the <i>TOP-List</i> of your website. If the hook is not set the record won't be listed in a top list. |
| Download-Counter | NOTICE: Field can be deactivated using the Plugin Configuration Options. This is a read only-field. It shows the number of times a record has been downloaded yet. It can be reseted to zero using the according button. |
| Filesize | The <i>file size</i> field is set automatically when a file is attached to this record. It contains the number of bytes the attached file has. This field cannot be edited. |
| Filetype | The <i>file type</i> field is set automatically when a file is attached to this record. It contains the content type of the attached file. This field cannot be edited. |
| Language | This field sets the language of the record. A record can be configured to be shown for all languages. Records that are translated have to belong to a language version with the type "[All]" of [default] |

| Fieldname | Function |
|----------------------|--|
| Original Translation | This field only appears when an non-default record is edited. It allows editors to attach the current record to an default record. |
| File category | Associates the record with one or more categories. |

Explaining the List-Types

The LIST-Mode

The LIST-Mode displays a list of items that are visible to the current user. The list can be configured to be sorted by the criteria *title*, *creation date*, *date of the last change*, *author* and *custom backend sorting*.

See chapter **Specific Sort Sequence** for more details about *custom backend sorting*.

The TOP-Mode

The TOP-Mode is a list of items that may be stored in maybe one hundred different folders. The extension takes all items that are visible to the current user and where the flag *'is VIP record'* is set and generates a list of them, sorted by the creation date of the list entries.

I needed this mode to select a few records out of a couple of folders and display them on the start page to have an eye catcher.

The PERSONAL-Mode

The PERSONAL-mode is a list of items that are stored in sys folders which each authorized user can combine after his personal needs. Therefore plugins using the LIST-Mode can be configured to show a button that allows user a add that plugin 's sys folders to his favorites.

This option is for the usage on big web projects, where several filelists on several pages receive their data from a lot of several sysfolders. User can subscribe items from the file lists they are interested in.

The SEARCH function

The SEARCH-mode displays a search bar. If someone searches the configured data source for a string, per default the fields *title*, *description* and *filename* are searched for the entered value.

The fields used in the search function can be set using TS.

The search results are shown in the list mode, so it uses the template- as well as the TypoScript parameters for the list mode.

The SINGLE-Mode

The SINGLE-Mode displays a single item. It doesn't make sense to have a page containing this mode standalone. It is useful to link from other pages containing the filelist-plugin to this page to display a record in a "sandbox".

Using the plugin

Specific Sort Sequence

Filelists configured to generate a simple list can be sorted after a few criteria, like the *title* or the *creation date*.

Specific Sort Sequence allows you to sort a list in a non-logical way. Therefore you've to configure the plugin to sort by *Custom Backend Sorting* in the field "Order by" this field (LIST-Mode).

This can look like this:

| Title | Author | Date | | |
|--------------------------------|---------|----------|--|-----------------------|
| Full Bundle | Jack | 23.03.06 | | [show] [download] |
| Only Component A | Adam | 25.03.06 | | [show] [download] |
| Only Component B | Simon | 01.03.06 | | [show] [download] |
| Component C+D | Jack | 01.04.06 | | [show] [download] |
| 30 days Evaluation Licence Key | Claudia | 09.03.06 | | [show] [download] |
| Readme | Adam | 28.02.06 | | [show] [download] |
| Installation Manual | Lisa | 24.03.06 | | [show] [download] |
| Installation Videomanual | Lisa | 24.03.06 | | [show] [download] |

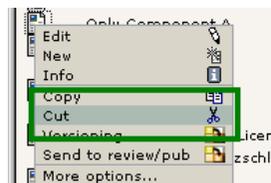
[Specific Sort Sequence]

The upstanding table has a custom order. This can be done by ordering the records in the Typo3 backend yourself:

Go to the list-mode, select the sys folder the records are stored in and order them in the same way as you would do it with page- or content objects using the clipboard.

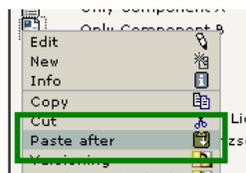


Click on the icon of the record you want to replace, select *Cut...*



and paste it onto its new position by selecting *Paste after*

The result...



| Title | Author | Date | | |
|--------------------------------|---------|----------|--|-----------------------|
| Full Bundle | Jack | 23.03.06 | | [show] [download] |
| Only Component A | Adam | 25.03.06 | | [show] [download] |
| Component C+D | Jack | 01.04.06 | | [show] [download] |
| Only Component B | Simon | 01.03.06 | | [show] [download] |
| 30 days Evaluation Licence Key | Claudia | 09.03.06 | | [show] [download] |
| Readme | Adam | 28.02.06 | | [show] [download] |
| Installation Manual | Lisa | 24.03.06 | | [show] [download] |
| Installation Videomanual | Lisa | 24.03.06 | | [show] [download] |

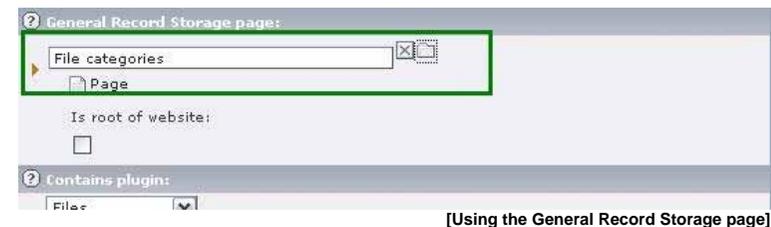
[Specific Sort Sequence]

Categories

The extension supports the use of categories. Therefore the sysfolder where records shall be created has to be configured from where to take the category records. This can be done in two different way, depending on the selection made in the Extension manager. You can either use the sysfolder's General Record Storage Path, or define a list of sysfolders from which the extension fetches the categories. See the Chapter Administration --> Installation for information on how to select the one of both solutions.

Using StoragePID

Edit the page details of the sysfolder where file records shall be created. Define the folder you want to be used as the storage of the categories in the field *General Record Storage page*.



[Using the General Record Storage page]

Using IDList

Edit the page details of the sysfolder where file records shall be created. Define a list of the PageIDs of the Sysfolders you want to be used as the storage of the categories in the field PageTS. Use the following line of TypoScript therefore:

```
TCEFORM.tx_drblob_content.category.PAGE_TSCONFIG_IDLIST = [Folder1], [Folder2], [...]
```

Use the PageID instead of [Folder1]. It is also valid to define only one Sysfolder here.



[Using IDList]

Configuration

HTML Templates

If you don't attach a template in the plugin configuration and don't enter a template file in the TypoScript Setup, the plugin uses some standard templates. See *Template File Priority* for additional information. These files are located in the folder `res` in the extension root directory. There is a template for every kind of view (list, top, single).

Template File Priority

There are 3 possible ways for the plugin to display a template: Selecting a template file using the plugin configuration (FlexForm), select a template using the TypoScript Setup, or leaving both blank to use the standard template files.

To avoid having template files I defined a flat hierarchy (from the high priority 1 to the low priority 3):

| Hierarchy | Template File |
|-----------|---|
| 1 | Template set in the plugin configuration |
| 2 | Template File set in the TypoScript Setup |
| 3 | Standard Templates |

If both (Plugin Configuration and TypoScript Setup) are left blank, the Extension uses the extensions standard Templates located in the `res`-Directory of the extension.

If a valid template file is entered in the plugin configuration, this file will be used. It will also overwrite the settings made in the TypoScript Setup.

HTML Example Template

There is also a file called `dr_blob.tpl`, which contains the default templates for all possible modes. I suggest you to build your templates by modifying this file. It is located in the sub folder `res` under the root directory of the extension, typically `typo3conf/ext/dr_blob/res`.

The SEARCH mode has its own template marker, called `###TEMPLATE_SEARCH###`. The results a search returns are displayed in the list mode (`###TEMPLATE_LIST###`), and so the template for list mode is also used. If a search returns no results the marker `###TEMPLATE_SEARCH_NOITEMS###` is used.

Template-Marker Reference

Template Subparts

| Subpart marker | Description |
|--|---|
| <code>###TEMPLATE_LIST###</code> | Template for the LIST mode. This mode is also used to display search results. |
| <code>###TEMPLATE_LIST_NOITEMS###</code> | Template for the LIST mode if a normal list does not contain any entries. |
| <code>###TEMPLATE_TOP###</code> | Template for the TOP-Mode |
| <code>###TEMPLATE_TOP_NOITEMS###</code> | Template for the TOP-Mode. Used if the list does not contain entries. |
| <code>###TEMPLATE_SINGLE###</code> | Template for a single item. |
| <code>###TEMPLATE_SEARCH###</code> | Template containing the search bar. |
| <code>###TEMPLATE_SEARCH_NOITEMS###</code> | Template to display the message that search did not return any records. |
| <code>###TEMPLATE_PERSONAL###</code> | Template for the PERSONAL-Mode |
| <code>###TEMPLATE_PERSONAL_NOITEMS###</code> | Template for the PERSONAL-Mode. Used if the list does not contain entries. |
| <code>###TEMPLATE_PERSONAL_ADD_FOLDER###</code> | Template for the button that allows users to add a folder to their favorites |
| <code>###TEMPLATE_PERSONAL_REMOVE_FOLDER###</code> | Template for the button that allows users to remove a folder from their favorites |

Template Marker

This chapter shows which markers are available for the four different template types (LIST, TOP, SINGLE, SEARCH).

LIST, PERSONAL, SINGLE, TOP

| Marker: | Description: | TOP | LIST | SINGLE | PERSONAL |
|--|--|-----|------|--------|----------|
| <code>###BLOB_TITLE###</code> <code>###LANG_TITLE###</code> | The marker contains the title of a record. | X | X | X | X |

| Marker: | Description: | TOP | LIST | SINGLE | PERSONAL |
|--|---|-----|------|--------|----------|
| <code>###BLOB_DESCRIPTION###</code> <code>###LANG_DESCRIPTION###</code> | The marker contains the description of a record. If the marker is used in a TOP- or in a PERSONAL-Element, the description will be shortened to the number of chars set with the TS Variable <code>plugin.tx_drblob.pil.topView. .lengthOfDescription</code> | X | X | X | X |
| <code>###BLOB_AUTHOR###</code> <code>###LANG_AUTHOR###</code> | The marker contains the author (cruser) of a record | X | X | X | X |
| <code>###BLOB_AUTHOR_EMAIL###</code> <code>###LANG_AUTHOR_EMAIL###</code> | The marker contains the email address of the author (cruser) of a record | X | X | X | X |
| <code>###BLOB_CRDATE###</code> <code>###LANG_CRDATE###</code> | The marker contains the create date of the record | X | X | X | X |
| <code>###BLOB_LASTCHANGE###</code> <code>###LANG_LASTCHANGE###</code> | The marker contains the date the record has been saved last. | X | X | X | X |
| <code>###BLOB_DOWNLOADCOUNT###</code> <code>###LANG_DOWNLOADCOUNT###</code> | The marker contains the number of times the according attached file was downloaded yet | X | X | X | X |
| <code>###BLOB_FILENAME###</code> <code>###LANG_FILENAME###</code> | The marker contains the filename of an attached file. | X | X | X | X |
| <code>###BLOB_FILESIZE###</code> <code>###LANG_FILESIZE###</code> | The marker contains the filesize in Bytes of an attached file. | X | X | X | X |
| <code>###BLOB_FILETYPE###</code> <code>###LANG_FILETYPE###</code> | The marker contains the content Type of the file | X | X | X | X |
| <code>###BLOB_FILEICON###</code> <code>###LANG_FILEICON###</code> | The marker contains an <code></code> -tag displaying the fileicon according to the filename of the attached file. The Extension uses (per default) the normal TYPO3 fileicons. | X | X | X | X |
| <code>###BLOB_SORTLINK_TITLE###</code> | The marker contains the List Header of the column <i>Title</i> wrapped in a sort Link | | X | | |
| <code>###BLOB_SORTLINK_CRDATE###</code> | The marker contains the List Header of the column <i>Create Date</i> wrapped in a sort Link | | X | | |
| <code>###BLOB_SORTLINK_LASTCHANGE###</code> | The marker contains the List Header of the column <i>LastChange</i> wrapped in a sort Link | | X | | |
| <code>###BLOB_SORTLINK_AUTHOR###</code> | The marker contains the List Header of the column <i>Author</i> wrapped in a sort Link | | X | | |
| <code>###BLOB_SORTLINK_FILESIZE###</code> | The marker contains the List Header of the column <i>Filesize</i> wrapped in a sort Link | | X | | |
| <code>###BLOB_SORTLINK_FILETYPE###</code> | The marker contains the List Header of the column <i>Filetype</i> wrapped in a sort Link | | X | | |
| <code>###BLOB_SINGLE_RTN_URL###</code> | The marker contains the URL that leads out of the single view | | | X | |
| <code>###BLOB_DATA_EXISTS_SWITCH_START###</code> | This marker provides the possibility to comment out some parts of the generated HTML code. If the current record does not have an attached file, this marker is set to <code><! - -</code> . It can for example be used to hide the fields <code>###BLOB_FILENAME###</code> or <code>###BLOB_FILESIZE###</code> . | | | X | |
| <code>###BLOB_DATA_EXISTS_SWITCH_END###</code> | This marker provides the possibility to comment out some parts of the generated HTML code. If the current record does not have an attached file, this marker is set to <code>- - ></code> . It can for example be used to hide the fields <code>###BLOB_FILENAME###</code> or <code>###BLOB_FILESIZE###</code> . | | | X | |
| <code>###BLOB_LINK_ITEM###</code> | Link wrapped in <code><a></code> -Tags to the single view of a record | X | X | | X |
| <code>###BLOB_LINK_FILE###</code> | Link wrapped in <code><a></code> -Tags to download a record | X | X | X | X |
| <code>###BLOB_MORE###</code> | The marker allows to insert a button to the single-view of a record (instead of having to wrap the title), like [more]. The value can be set via TS. The advantage of using this marker is that it is translated according to the page lang. | X | X | | X |
| <code>###BLOB_DOWNLOAD###</code> | The marker allows to insert a button to download a record (instead of having to wrap the title), like [download]. The value can be set via TS. The advantage of using this marker is that it is translated according to the page lang. | X | X | X | X |
| <code>###LANG_NOITEMS###</code> | No-Items-Found-Message | X | X | X | X |
| Deprecated Marker | | | | | |

| Marker: | Description: | TOP | LIST | SINGLE | PERSONAL |
|--------------------------|---|-----|------|--------|----------|
| ###BLOB_MORE_LINK### | Deprecated! The marker contains the button to the single view of a record. The List Mode provides the possibility to set the value of the <i>more-link</i> via TypoScript. | X | X | | X |
| ###BLOB_TITLE_LINK### | Deprecated! The marker contains the title of a record wrapped in a link to this item | X | X | | X |
| ###BLOB_DOWNLOAD_LINK### | Deprecated! The marker contains the button to download a file. The List Mode provides the possibility to set the value of the download link via TypoScript. | X | X | X | X |

SEARCH

| Marker: | Description: |
|---------------------|--|
| ###FORM_URL### | This marker contains the URL the form is sent to. |
| ###SWORDS### | This marker contains the word the user searched for. |
| ###SEARCH_BUTTON### | This marker contains the Local Language String for the button's title. |

PERSONAL_ADD / PERSONAL_REMOVE

| Marker: | Description: |
|-------------------|---|
| ###FORM_METHOD### | This marker contains the method the form uses send its data. |
| ###FORM_TARGET### | This marker contains the URL the form is sent to. |
| ###ACTION### | Marker containing the values of the field <i>action</i> (IMPORTANT) |
| ###ITEMS### | Marker containing the values of the field <i>items</i> (IMPORTANT) |

TypoScript Reference

Different Options for different modes

We have to differ between options in extension-global scope, and options for a special Mode (or View).

| TS Option: | Scope |
|--------------------------------------|--------------------------------|
| plugin.tx_drblob_pi1 {} | Extension-Global settings |
| plugin.tx_drblob_pi1.listView {} | Settings for the LIST-Mode |
| plugin.tx_drblob_pi1.personalView {} | Settings for the PERSONAL-Mode |
| plugin.tx_drblob_pi1.singleView {} | Settings for the SINGLE-Mode |
| plugin.tx_drblob_pi1.topView {} | Settings for the TOP-Mode |

Extension Global Settings

The example TypoScript Template has to be included as static. Use the "Include static (from extensions)" feature in your template record.

| Property: | Data type: | Description: | Default: |
|--------------|------------|--|---------------------------|
| templateFile | String | This is the option to set if you like to define one template for multiple dr_blob plugins in your page tree . See Configuration --> <i>Templates</i> for additional information. Example: <pre>plugin.tx_drblob_pi1 { templateFile = fileadmin/template.tpl }</pre> | <i>Reference Template</i> |

| | | | |
|--------------------|---------|---|---|
| fileExtIconFolder | String | Use this Option to define a custom folder containing the file extension icons. The files have to be named as follows: <code>[EXTENSION].gif</code> If no icon is found for the current file extension the image <code>default.gif</code> is used. Example: <pre>plugin.tx_drblob_pi1 { fileExtIconFolder = fileadmin/icons/ }</pre> | typo3/sysext/cms/tslib/media/fileicons/ |
| searchFieldList | String | Comma-separated string of the fields the search function searches. Possible fields are <code>title</code> , <code>description</code> and <code>blob_name</code> Example: <pre>plugin.tx_drblob_pi1 { searchFieldList = title,description emptySearchAtStart = 0 }</pre> | <i>All possible fields</i> |
| emptySearchAtStart | Boolean | If set the SEARCH will show the search box at start. Otherwise a full list of items is shown. | 1 |

TS Options for all Modes

| Property: | Data type: | Description: | Default: |
|------------------|------------|---|---------------------------------|
| date_stdWrap | ->stdWrap | Here you can set the date formatting for the template marker ###BLOB_CRDATE### and ###BLOB_LASTCHANGE### using a stdWrap. Example: <pre>plugin.tx_drblob_pi1.listView { #May use 'date'... date_stdWrap.date = d.m.Y #...or 'strftime'... date_stdWrap.strftime = %d.%m.%Y }</pre> | m/d/Y stdWrap of Type 'date' |
| email_stdWrap | ->stdWrap | Formats the email address of the author of a record using a stdWrap. The value parameter will be overwritten. Example: <pre>plugin.tx_drblob_pi1.singleView { email_stdWrap.typolink.ATagParams = class='YourCSS' }</pre> | |
| filesize_stdWrap | ->stdWrap | Formats the filesize of a record using a stdWrap. Example: <pre>plugin.tx_drblob_pi1.singleView { filesize_stdWrap.bytes = 1 filesize_stdWrap.bytes.labels = B KB MB GB }</pre> | 1 B KB MB GB |
| categoryDivider | String | This value appears between the category titles shown in the records. | , (Comma) |
| category_stdWrap | ->stdWrap | stdWrap for the list of categories. Example: <pre>plugin.tx_drblob_pi1.listView { category_stdWrap.wrap = }</pre> | |

Special Options for the ListView, the Personal-View, and the Top-View

The Options listed below are modified using the TS Object according to the mode:

| Property: | Data type: | Description: | Default: |
|---------------------|------------|---|----------|
| alternatingLayouts | Integer | Indicates how many alternating designs the script should expect in the html template (compare with <code>tt_news</code> where I got the idea of providing multiple layouts) Example: If you define a subpart like: " <code><!--##BLOBITEM##--> ... <!--##BLOBITEM##--></code> " this is used all the time. If you define a similar subpart: " <code><!--##BLOBITEM_1###--> ... <!--##BLOBITEM_1###--></code> " which might show another set of colors, this is used every second time instead of the default! This is because "alternateLayouts" is set to 2. If you define a similar subpart " <code><!--##BLOBITEM_2###--> ... <!--##BLOBITEM_2###--></code> " ... this will be used every third time IF (!) "alternateLayouts" is set to 3. If you do not set it to 3, the first two alternating designs will be used only. | 2 |
| lengthOfDescription | Integer | Set the number of chars the preview of the description-field contains. | 150 |

[TS for the LIST-, the PERSONAL- and the TOP-Mode]

Example

```
plugin {
  tx_drblob_pil {
    templateFile = fileadmin/SITE/assets/tmpl/template.tpl
    fileExtIconFolder = fileadmin/SITE/assets/fileicons/
    searchFieldList = title,description
    emptySearchAtStart = 1

    listView {
      date_stdWrap.date = d.m.Y
      category_stdWrap = |
      categoryDivider = ,
      alternatingLayouts = 2
      lengthOfDescription = 150
    }

    singleView {
      date_stdWrap.strftime = %d.%m.%Y
      category_stdWrap.wrap = <ul><li></li></ul>
      categoryDivider = </li><li>
      email_stdWrap {
        typolink {
          ATagParams = class='emailLinkCSS'
          ATagBeforeWrap = 1
        }
        wrap = |
      }
    }

    topView {
      lengthOfDescription = 250
      date_stdWrap.date = d.m.Y H:i
      alternatingLayouts = 2
      categoryDivider = &nbsp;
    }

    personalView {
      lengthOfDescription = 250
      date_stdWrap.date = d.m.Y H:i
      alternatingLayouts = 2
      categoryDivider = &nbsp;
    }
  }
}
```

Tutorial

Customizing [more] & [download]-links

This chapter describes how to customize the link to the single view and the link to download a file.

Templates

The frontend-layout of the extension can (as you know) be customized using the html-templates rolled out with the extension.

Lets have a look at the markers used there:

```
BLOB_MORE
BLOB_DOWNLOAD
LINK_ITEM
LINK_FILE
```

The markers `LINK_` contain `<a>`-Tags that link either to the single view or to a file. They have to wrap the text you want to use as the link title, for example `<!--##LINK_FILE##-->[get File]<!--##LINK_FILE-->` will be parsed to `[get File]`

For multi-language-support you should use the marker `BLOB_MORE` or `BLOB_DOWNLOAD`. These values are parsed using the Typo3 Frontend Localization Engine. The localized content can be manipulated using TS.

Typo Script

You can define the value used for the [more]- and the [download]-Button using the following TS-marker:

```
plugin.tx_drblob_pil._LOCAL_LANG.[LANG].list_button_download = VALUE
plugin.tx_drblob_pil._LOCAL_LANG.[LANG].list_button_show = VALUE
plugin.tx_drblob_pil._LOCAL_LANG.[LANG].personal_button_download = VALUE
plugin.tx_drblob_pil._LOCAL_LANG.[LANG].personal_button_show = VALUE
plugin.tx_drblob_pil._LOCAL_LANG.[LANG].single_button_download = VALUE
plugin.tx_drblob_pil._LOCAL_LANG.[LANG].top_button_download = VALUE
plugin.tx_drblob_pil._LOCAL_LANG.[LANG].top_button_show = VALUE
```

Example

```
plugin {
  tx_drblob_pil {
    _LOCAL_LANG {
      de {
        list_button_show = 
        personal_button_show = [show me more]
        top_button_show = plugin.tx_drblob_pil._LOCAL_LANG.de.personal_button_show
      }
    }
  }
}
```

Abandoned Feature for the LIST-Mode

In the LIST-mode there is a feature that is not needed anymore but may be helpful. There is no guaranty the it is also a part of future versions.

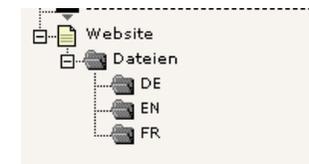
The feature allows you to set the buttons for single view and download globally, no matter what language is used.

```
plugin.tx_drblob_pil.listView.showButtonValue = { more }
plugin.tx_drblob_pil.listView.downloadButtonValue = { download }
```

Having an own sysfolder for each language

I suggest using the Typo3 Inbound Localization Features, the method described here is just a workaround!

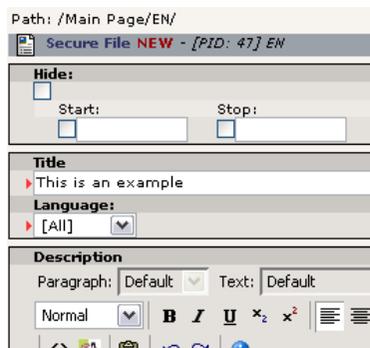
Some people ask how to configure the plugin to use an own sysfolder for each language. They don't want to use the inbound localization features, whyever. Their page tree could look like that...:



[An own Sysfolder for each language]

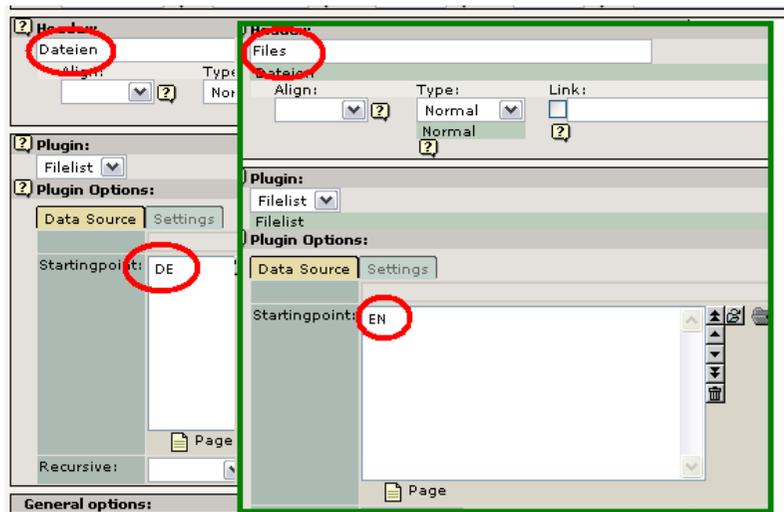
If you simply create new objects, the database-column `sys_language_uid` will be set to 0. This equals the default frontend language, so the created documents are only shown when the default frontend language is requested.

The easiest way to workaround this is to set the language-flag for each record to [ALL], so the record is shown no matter which language is requested.



[Set language to [ALL]]

Set the page in the plugin configuration to the according sysfolder, and do that for each language you want to use:

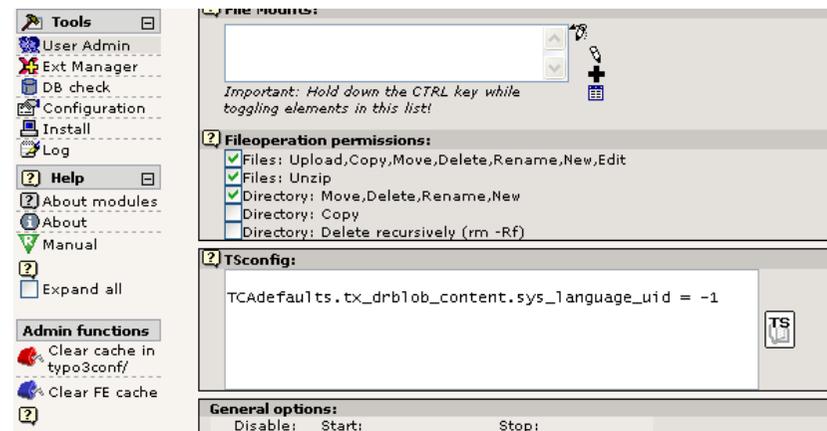


[An own Sysfolder for each language: Plugin Configuration]

User TS Config

A little assistance may be to preselect the language field so that you don't have to change it everytime you create a record. Therefore add the following line of TypoScript to an user's or a group's User TS-Config-Field:

[Setting the default language flag to "[ALL]"]



Simply add for following line

```
TCAdefaults.tx_drblob_content.sys_language_uid = -1
```

This will set the language field to 'ALL' per default.

Known problems

- Apart from my bad English ... Nothing @ the moment :-)

To-Do list

High priority

- Implement the possibility to store files in the file system too.
- Render all fields using a stdWrap

Medium priority

- Making a attached file deletable
- Add a page browser

Changelog

| Version | Changelog | Date |
|---------|--|------------|
| 1.0.x | First published Version | 10-20-2005 |
| 1.1.x | Features: <ul style="list-style-type: none"> Added the Search function Bugfixes: <ul style="list-style-type: none"> Some bugfixes | 10-27-2005 |
| 1.2.x | Features: <ul style="list-style-type: none"> Implemented the Typo3 Localization Features Added a cache control header for downloadable files Added the TypoScript Variable to set a global template as well as the template file priority. | 11-29-2005 |
| 1.3.x | Features: <ul style="list-style-type: none"> Added the possibility to display fileicons according to the attached files Added the calculation for the filesize Did some changes in the record input mask Prepared for the versioning used by workspaces | 12-12-2005 |
| 1.4.0 | Features: <ul style="list-style-type: none"> Added the personalizable list as well as the according configuration buttons and the flexform value. Added the template marker to display the number of times a record was downloaded yet Bugfixes: <ul style="list-style-type: none"> Fixed a bug that occurred after attaching a file and pressing the Save&New Button and caused the loss of the file. Fixed a bug that occurred in IE6 when sending Cache Control Headers. The downloaded file was not readable | 03-21-2006 |

| Version | Changelog | Date |
|---------|---|------------|
| 1.5.0 | <p>Features:</p> <ul style="list-style-type: none"> Added the TypoScript-Marker to define a folder for custom file extension-icons. Changed the date_stdWrap-Option in TS. „date_stdWrap“ now is a real stdWrap. Added the possibility to reset the download counter of an attached file. Added specific sort sequence to the list mode. Added the field-change-function to the custom TCA fields. <p>Changes:</p> <ul style="list-style-type: none"> Migrated the locallang-files to the <T3locallang> XML-data-structure. Moved the definition of the custom input elements to an own class to avoid a naming conflict. Replaced the XCLASS through a hook on TCEmain <p>Bugfixes:</p> <ul style="list-style-type: none"> Fixed the bug of the file-icons being displayed from the wrong folder (the one from T3 3.x) Enabled overwriting LocalLang-Labels via the TS Object [. . .] . _LOCAL_LANG. Now hiding the default-language-version of localized records. | 10-04-2007 |
| 1.5.1 | <p>Features:</p> <ul style="list-style-type: none"> Added the template-markers BLOB_MORE, BLOB_DOWNLOAD, LINK_ITEM & LINK_FILE Added the foldericon <p>Changes:</p> <ul style="list-style-type: none"> Changed fieldnames in FE locallang to enable them to be set via TS The template-markers BLOB_TITLE_LINK, BLOB_MORE_LINK & BLOB_DOWNLOAD_LINK are deprecated The fields "sys_langugage_uid" and "t18n_parent" are now changeable using TCA <p>Bugfixes:</p> <ul style="list-style-type: none"> Re-wrote parts of the localization features. They finally should work now ;-) | 04-05-2007 |
| 1.6.0 | <p>Features:</p> <ul style="list-style-type: none"> Added the categorization feature Added the TS config parameter to set the fields used by the search function Added Template Marker for Localization of field names. They use the namespace ###LANG_[field]### The eMail-address-marker now uses a cObject to allow spam protection of the email addresses. <p>Changes:</p> <ul style="list-style-type: none"> Removed CSS Classes that didn't use the namespace tx_drblob_ from the reference template The example extension template has to be included static now. <p>Bugfixes:</p> <ul style="list-style-type: none"> Removed a bug preventing the PERSONAL-mode from wrapping the date correctly. | 06-14-2007 |
| 1.6.1 | <p>Features:</p> <ul style="list-style-type: none"> The eMail Link is now configurable using a stdWrap The filesize is now configurable using a stdWrap Added a SORTLINK-Marker for the filesize (blob_size) and filetype (blob_type) <p>Changes:</p> <ul style="list-style-type: none"> Removed the TS-Values fileExtIconHeight and fileExtIconWidth. They are now detected automatically Changed the is_vip – field from type "enum" to type "tinyint" to increase dbal-compatibility Changed the field blob_size from varchar to int to allow sorting | 09-25-2007 |

Credits

Thanks to...

- Martin Kutschker for giving me good advises for the stdWraps of date-Fields
- Martin Kutschker and Thomas van de Pas for suggesting the specific sort sequence.
- Christoph Lorenz for reporting and helping me to bugfix the bug of the Cache Control Header in combination with Internet Explorer 6
- All users for using this extension and reporting bugs