Browser Tutorial Search and Filter (en)

Provide a search form and filters for your data on your website. The Browser - the TYPO3-Frontend-Engine - enables searching and filtering in every database. The Browser enables to search in areas like from last month to next month or from 10.000 EUR to 20.000 EUR.

Version: 3.6.0, 2010-12-29 - Version is corresponding with the version of the Browser.

Extension Key: browser_tut_search_en - Tutorial for the extension Browser (extkey: browser)

Language: en

Keywords: forAdmins, forDevelopers, forBeginners, forAdvanced, browser, tutorial, frontend-engine, frontend, engine, search

Copyright 2009-2010, Dirk Wildt, Die Netzmacher, <http://wildt.at.die-netzmacher.de>

The content of this document is related to TYPO3

- a GNU/GPL CMS/Framework available from www.typo3.org

Table of Contents

Screen Shots

Juridat

Three screen shots with the extension juridat and an optimized style sheet.

Illustration 1: List view

Illustration 2: Result with six search words

Illustration 3: Result in the single view

tt_news

Three screen shots with the extension tt_news without any optimized style sheet.

Illustration 4: List view
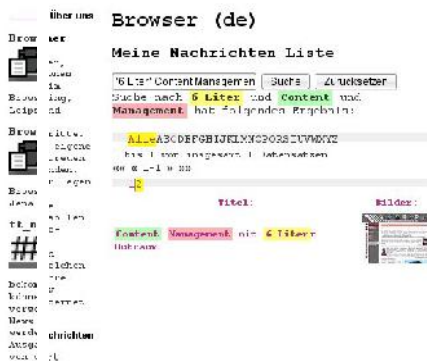

Illustration 5: Result with phrases and words

Illustration 6: Result in the single view

Green Cars

Example with dynamic filters

Illustration 7: Green Cars with a selection example

Illustration 8: Each filter controls all other filters.

See the tutorial 'Green Cars - Dynamic Filter' on page 20 below.

Introduction

What does the Tutorial do?

This is a tutorial for the extension Browser (extkey: browser).

It is only

- a step-by-step introduction for the configuration of the search functionality and filters.

If you are interested in other issues, please note the section "Further Informations" below.

What does the Browser Search and Filter do?

- Searching in displayed and in non displayed columns.

- Searching for words and phrases. Example:

  - Sword: "My Phrase" Word another word

  - Will search for "My Phrase" or "Word" or "another" or "word".

- The search is case insensitive.

- The search is configurable:

  - You can search in every column of every table. It doesn't matter, if table or field are displayed in the current view.

  - You can configure the search form and the colours of the result items among others.

- Filter your data by checkboxes, radiobuttons and selectboxes

  - Filter are configurable. Examples for possibilities:

    - Items has the stdWrap property

    - You can add a fist item like "All Cars"

  - Filter are dynamically. If there isn't any hit, filter items won't be displayed. It's configurable.

  - Filter count hits. It's configurable.

  - Filter can handle data areas. Example for a data area: from January to December, from 10.000 EUR to 20.000 EUR
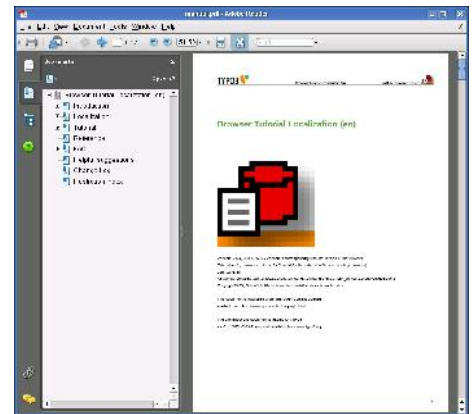
Illustration 9: Tutorial in PDF format

Further Information

This tutorial in PDF format

You find this manual as PDF file at

- doc/manual.pdf

- http://typo3.org/extensions/repository/view/browser_tut_search_en/current/info/?tx_terfe_pi1%5BdownloadFile%5D=doc%252Fmanual.pdf

Browser Manual and Browser Tutorial Basics

Manual: The Browser has it's own manual (with the reference of course).

- HTML http://typo3.org/documentation/document-library/extension-manuals/browser/current/

- PDF: http://typo3.org/extensions/repository/view/browser/current/info/?tx_terfe_pi1%5BdownloadFile%5D=doc%252Fmanual.pdf

Tutorial Basics: A lot of step-by-step introductions.

- HTML: http://typo3.org/documentation/document-library/extension-manuals/browser_tut_basics_en/current/

- PDF: http://typo3.org/extensions/repository/view/browser_tut_basics_en/current/info/?tx_terfe_pi1%5BdownloadFile%5D=doc%252Fmanual.pdf

Tutorial Search

Example with tt_news

Requirements

You need

- the extension Browser of course
- the extension tt_news[1]
- a page with
  - the plugin browser
    - set in the tab [General] the type of the header to hidden
    - set in the tab [Plugin] the startingpoint to your sysfolder with the news records
  - an extension template
    - with the include static template of the browser
    - and in the setup field the TypoScript from below.

TypoScript

```
plugin.tx_browser_pi1 {
  views {
    list {
      1 {
       select = tt_news.title, tt_news_cat.title
      }
    }
    single {
      1 {
       select = tt_news.title, tt_news.short, tt_news.bodytext, tt_news_cat.title
      }
    }
  }
}
```

This is the TypoScript for

- a list view with the fields tt_news.title and tt_news_cat.title and
- a single view with the fields tt_news.title, tt_news.short, tt_news.bodytext, tt_news_cat.title.

Result in the Frontend

The Result in the frontend should look like the screen shot below, if you are using the German language:

Illustration 10: Frontend with a simple example with tt_news

---

1   You can take every other extension, but than you need a TypoScript with its tables and fields instead of tt_news.

Test it!

Familiarize you with the search.

Example with two words

Search for the words: Browser Erweiterung

Illustration 11: List view with marked search words

Illustration 12: Single view with marked search words

Example with seven words

Search for seven words: Browser Erweiterung Bequem durch die Daten browsen

Illustration 13: List view with marked search words

Illustration 14: Single view with marked search words

With words and phrases

Search for words and phrases: Browser Erweiterung "Bequem durch die Daten browsen"

Illustration 15: List view with marked words and phrases

Illustration 16: Single view with marked words and phrases

Configuring the Search Form

Configuring with the Plugin

Illustration 17: The Browser plugin. Tab [List] (1) with the search menu (2)

1. Plugin Tab [Lists]

2. Set the Search mode to [Configured]

Comment

- Search: Tutorials
  Dummy check box for information only. link to the online tutorial

- Search: Form

    - [X] Display the form

    - [X] with result phrase

Result phrase will appear below the form.

- [X] with colored swords (list view)
  Search words will be colored in the list view. But there is an exclusion: Search words linked values won't be colored.

- [ ] with colored swords (single view)
  Search words will be colored in the single view. But there are some exclusions: Search words in children records and linked values won't be colored.

- Search: Checkpoint for colored search words in the single view
  Dummy check box for information only.
  If you have the search word in the URL, you can get different links to the same content. Search engines doesn't like it!

- Search: wild-cards

  - Self-acting (recommended)
    Search words will wrapped with wild-cards by default

  - User have to add wild-cards

- Search: char for a wild-card
  You can use another wild-card. Be aware that are conflicts with the SQL queries: '%' isn't possible for example.

List without any Search Form

Illustration 18: List view with the search form

Illustration 19: List view without any search form

Search Form without any Result Phrase

Illustration 20: Search form with result phrase

Illustration 21: Search form without any result phrase

Configuring the Result Phrase

If you want change the phrase of the result phrase, please configure

```
plugin.tx_browser_pi1.displayList.display.searchform.resultPhrase {
  searchFor.value = Search for
  searchFor.lang.de = Suche nach
  ...
}
```

Search results with other Colors

If you like to want less, more or other colors for the search result, please configure the array wrapSwordInResult.

Be aware that displayList and displaySingle have there own search form array.

```
// List View

plugin.tx_browser_pi1.displayList.display.searchform {
  searchform {
    wrapSwordInResults {
      0 {
        wrap = <span style="background:#ffff7f">|</span>
      }
      1 {
        wrap = <span style="background:#b3ffb2">|</span>
      }
      ...
    }
  }
}


// Single View

plugin.tx_browser_pi1.displaySingle.display.searchform {
  searchform {
    wrapSwordInResults {
      0 {
        wrap = <span style="background:#ffff7f">|</span>
      }
      1 {
        wrap = <span style="background:#b3ffb2">|</span>
      }
      ...
    }
  }
}
```

Extending the Search

Usually visitors of a website can search in the displayed columns only.

The Browser enables to search in every table and field you like - independing of the display mode.

Example

You want to display in the list view only the fields: tt_news.title, tt_news_cat.title.

But the visitor should search in the fields: tt_news.title, tt_news.short, tt_news.bodytext.

TypoScript

```
plugin.tx_browser_pi1 {
  views {
    list {
      1 {
        select = tt_news.title, tt_news_cat.title
        search = tt_news.title, tt_news.short, tt_news.bodytext
      }
    }
  }
}
```

Search and Filter

By default the search doesn't respect any filter. This has an effect only, if you are using filters or a category menu.

The effect is, that

Security

The Minimum Length of a Sword Word

The default for the minimum length of a sword word is three chars. If a word is shorter, it will be removed from the sword.

If you want to increase or decrease the minimum length, please configure minLenWord:

```
plugin.tx_browser_pi1.advanced {
  security {
    sword {
      minLenWord = 3
    }
  }
}
```

MySQL Security: add Slashes

The Browser secures swords with the PHP method mysql_real_escape_string().

mysql_real_escape_string() adds slashes to \x00, \n, \r, \, ', " and \x1a, but not to \%.

The Browser adds a slash to \% too. If you don't like it, please configure csvChars:

```
plugin.tx_browser_pi1.advanced {
  security {
    sword {
      addSlashes {
        csvChars = %
      }
    }
  }
}
```

If you need slashes for more than the \%, please add the chars to csvChars comma separated like in the code below:

```
plugin.tx_browser_pi1.advanced {
  security {
    sword {
      addSlashes {
        csvChars = %, `, ´
      }
    }
  }
}
```

Tutorial Filters and a Category Menu

You can extend the search form with filters in form of

- checkboxes,
- radiobuttons and
- selectboxes.

And you can add a

- category menu

to your HTML template within or without the search form.

The Common Ground

Both have the same Effect

There is no difference between filters and a category menu in principle. Both enables the visitor of your website to search structured. Structured means:

Visitors can limit the amount of records – they can search for data – by given categories like:

- Display all news with the category "society".
- Display all events, which will happen "next week".

The difference between filters and a category menu isn't the effect but the rendered HTML code.

The Difference

Filters are a Part of an HTML Form

Filters have to be rendered as a COA object CHECKBOX, RADIOBUTTON or SELECTBOX. All of them are a part of an HTML form. Non item of them will be rendered as a link directly. All of them have to be placed within the Marker ###SEARCHBOX###.

AJAX

Because of AJAX filters have to stand outside the <div class="listarea">.

A Category Menu is a List of Links

But every items of a CATEGORY_MENU will be rendered as a link like the items of your main menu. A CATEGORY_MENU isn't a part of a form. A CATEGORY_MENU has to be placed within the marker ###CATEGORY_MENU###.

AJAX

Because of AJAX filters have to stand inside the <div class="listarea">.

Disabled by default

The subpart ###CATEGORY_MENU### is disabled by TypoScript by default. The subpart will removed from the HTML template – you won't have any waste in your code.

If you are using the category menu please enable in your global or local configuration:

```
displayList.display.category_menu = 1
```

Workflow

1. Add to your template the table.field, which should used as a filter or category menu
    1. You have to place it within the ###SEARCHBOX### marker, if you like to use a filter as a checkbox, radiobutton or selectbox.
    2. You have to place it within the ###CATEGORY_MENU### marker, if you like to use a filter as a category menu.

2. Add to your TypoScript the table.field

3. Copy one of the Content Objects CATEGORY_MENU, CHECKBOX, RADIOBUTTON or SELECTBOX from the master_templates array to the table.field.

4. Configure the category menu , checkbox, radiobuttons or selectbox.

A good example for filter is the extension Green Cars. Green Cars provides the German case story "Fallstudie grüne-autos.org - Datenbank in drei Stunden". It explains very well, what you have to know, if you want to use filters.

Be aware that you can filter data of the local table and foreign tables – all tables – from version 3.6.0.

See

- Extension Green Cars in the TYPO3 repository
  http://typo3.org/extensions/repository/view/green_cars/current/

- Fall Studie "Grüne Autos"
  http://typo3.org/extensions/repository/view/green_cars/current/info/?tx_terfe_pi1%5BdownloadFile%5D=doc%252Ffall_studie.pdf

Illustration 22: http://gruene-autos.org/ with three selectboxes

HTML Template

Add to your template the table.fields you want to use as filters like in the snippet below.

See line 10 to 12.

```
1.  <!-- ###TEMPLATE_LIST### begin -->
2.      ###MY_TITLE###
3.  <!-- ###SEARCHFORM### begin -->
4.          <div class="searchbox searchbox-###VIEW###-###MODE###">
5.            <form action="###ACTION###" method="post" name="Testform">
6.              <input type="hidden" name="no_cache" value="1" />
7.              <input class="sword" type="text" name="tx_browser_pi1[sword]"
                  onfocus="if(this.value =='###SWORD_DEFAULT###')
                  this.value=''" value="###SWORD###" />
8.              <input class="button" type="submit" value="###BUTTON###" />
9.              <input class="button" type="button" onclick="location='###MY_URL###'"
                  value="###MY_RESET###" />
10.             ###TX_BZDSTAFFDIRECTORY_LOCATIONS.TITLE###
11.             ###TX_BZDSTAFFDIRECTORY_FUNCTIONS.TITLE###
12.             ###TX_BZDSTAFFDIRECTORY_GROUPS.GROUP_NAME###
13.             ###RESULTPHRASE###
14.           </form>
15.         </div><!-- ###SEARCHFORM### end -->
16.         ...
17. ...
```

TypoScript

Add the table.fields like in the snippet below.

Line 6: Add the array filter.

Line 7, 10, 13: Add the tables.
Line 8, 11, 14: Add the fields.
qLine 8, 11, 14: Copy to the fields the selectbox from the array master_templates

```
1.  plugin.tx_browser_pi1 {
2.    views {
3.      list {
4.        1 {
5.          ...
6.          filter {
7.            tx_bzdstaffdirectory_locations {
8.              title < plugin.tx_browser_pi1.displayList.master_templates.checkbox
9.            }
10.           tx_bzdstaffdirectory_groups {
11.             group_name < plugin.tx_browser_pi1.displayList.master_templates.radiobuttons
12.           }
13.           tx_bzdstaffdirectory_functions {
14.             title < plugin.tx_browser_pi1.displayList.master_templates.selectbox
15.           }
16.         }
17.         ...
18.       }
19.     }
20.   }
21. }
```

TypoScript Object Browser
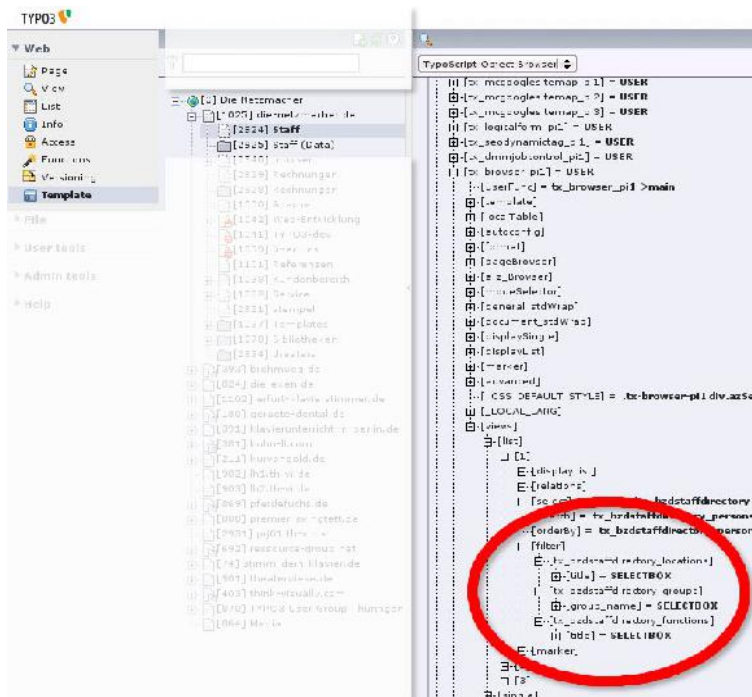
Illustration 23: The object SELECTBOX

If you want to configure a filter / select box, please open the field array in your TypoScript Object Browser.

You will see a lot of commented properties like in the illustration below:
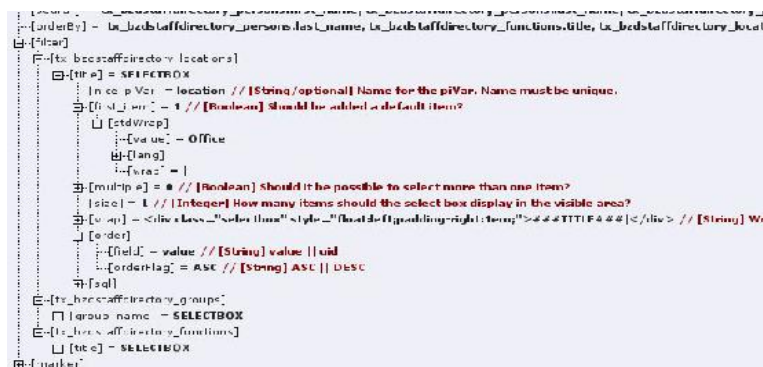
Illustration 24: The object SELECTBOX with some properties.

That's it!

Examples

Quick Shop - 1 Filter

It is very simple to configure the content objects

- SELECTBOX
- CHECKBOX and
- RADIOBUTONS

All have a master template. You copy the master template to your view and you configure the properties only, which aren't configured for your needs.

If you are interested in the power of a master template, please look at

- Reference 'SELECTBOX' on page 30 below

After copying you will see the powerful TypoScript code supported by the TypoScript Object Browser in your view array. It should be comprehensible, how to change any feature and property of your filter to your needs.

See section

- 'TypoScript Object Browser' on page 15 above and
- Reference 'master_templates' on page 28 below

See the examples below.

SELECTBOX

Illustration 25: Quick Shop with the filter SELECTBOX

See the TypoScript on the next page.

TypoScript

See the line

- // The filter becomes the property SELECTBOX

below in the snippet

```
plugin.tx_browser_pi1 {
  ...
  views {
    list {
      1 {
        select = ...
        filter {
          tx_quickshop_categories {
            // The filter becomes the property SELECTBOX
            title < plugin.tx_browser_pi1.displayList.master_templates.selectbox
            title {
              // Configuration SELECTBOX begin
              first_item {
                stdWrap {
                  data >
                  value = Select a Category
                  lang {
                    de = Kategorie wählen
                  }
                }
              }
              wrap = <span class="selectbox">|</span>
              wrap {
                item {
                  stdWrap {
                    crop = 60 | ... | 1
                  }
                }
              }
              // Configuration SELECTBOX end
            }
          }
        }
      }
    }
    single {
      ...
    }
  }
}
```

Easy way of configuring: See section 'TypoScript Object Browser' on page 15 above.

CHECKBOX

Illustration 26: Quick Shop with the filter CHECKBOX

TypoScript

See the line

- // The filter becomes the property CHECKBOX

below in the snippet

```
plugin.tx_browser_pi1 {
  ...
  views {
    list {
      1 {
        ...
        filter {
          tx_quickshop_categories {
            // The filter becomes the property CHECKBOX
            title < plugin.tx_browser_pi1.displayList.master_templates.checkbox
          }
        }
      }
    }
  }
}
```

RADIOBUTTONS

Quick Shop

Kategorie
◯ Kleidung (4) ◯ Becher (1) ◯ Bücher (1)

Begriff suchen   Suche Zurücksetzen

Alle|A**B**|C|D|E|F|**G**|H|I|J|K|L|M|N|O|**PQ**|R|S|**T**|U|V|W|XYZ

1 bis 3 von insgesamt 6 Datensätzen
«« « 1-3 **4-6** » »»

Produkt ▾

Pullover

Rot mit TYPO3 Kopf.

Geheime Bilder

Illustration 27: Quick Shop with the filter RADIOBUTONS

TypoScript

See the line

- // The filter becomes the property RADIOBUTONS

below in the snippet

```
plugin.tx_browser_pi1 {
  ...
  views {
    list {
      1 {
        ...
        filter {
          tx_quickshop_categories {
            // The filter becomes the property RADIOBUTONS
            title < plugin.tx_browser_pi1.displayList.master_templates.radiobutons
          }
        }
      }
    }
  }
}
```

Green Cars - Dynamic Filter

The Browser filters are dynamically by default. This means:

- If a filter item hasn't any hit, it won't be displayed.
- Each filter controls all others..

Exercise

We will change some TypoScript of the page http://gruene-autos.org/ for demonstration only.

See the examples below.

SELECTBOX (Default)

Illustration 28: Green Cars with three selectboxes - each with size = 1

This is the start page of http://gruene-autos.org/

It has three filters designed as selectboxes. Each displays one item only (HTML property size is 1).

The TypoScript for the three selectboxes should be clear (see the snippet below).

TypoScript

See the lines

- // This filter becomes the property SELECTBOX

below in the snippet

```
plugin.tx_browser_pi1 {
  ...
  views {
    list {
      1 {
        select = ...
        filter {
          tx_greencars_engine {
            // This filter becomes the property SELECTBOX
            title < plugin.tx_browser_pi1.displayList.master_templates.selectbox
            title {
              first_item {
                stdWrap {
                  data = LLL:EXT:green_cars/locallang_db.xml:tx_greencars_main.engine
                }
              }
              wrap = |
            }
          }
          tx_greencars_pricebracket {
            // This filter becomes the property SELECTBOX
```

```
                title < plugin.tx_browser_pi1.displayList.master_templates.selectbox
                title {
                  first_item {
                    stdWrap {
                      data = LLL:EXT:green_cars/locallang_db.xml:tx_greencars_main.pricebracket
                    }
                  }
                  wrap = |
                  order {
                    field = uid
                  }
                }
              }
              tx_greencars_type {
                // This filter becomes the property SELECTBOX
                title < plugin.tx_browser_pi1.displayList.master_templates.selectbox
                title {
                  first_item {
                    stdWrap {
                      data = LLL:EXT:green_cars/locallang_db.xml:tx_greencars_main.type
                    }
                  }
                  wrap = |
                }
              }
            }
          }
        }
      }
    }
```

SELECTBOXES size = 5

For a better understanding of dynamic filters we change the properties of each selectbox. We like to display five items per box and we remove the title (the first item). See the illustration below.

Illustration 29: Green Cars with three selectboxes - each with size = 1

TypoScript

```
// Set the size of all boxes to 5
plugin.tx_browser_pi1.views.list.1.filter.tx_greencars_engine.title.size = 5
plugin.tx_browser_pi1.views.list.1.filter.tx_greencars_pricebracket.title.size = 5
plugin.tx_browser_pi1.views.list.1.filter.tx_greencars_type.title.size = 5
// Don't display the first_iem
plugin.tx_browser_pi1.views.list.1.filter.tx_greencars_engine.title.first_item = 0
plugin.tx_browser_pi1.views.list.1.filter.tx_greencars_pricebracket.title.first_item = 0
plugin.tx_browser_pi1.views.list.1.filter.tx_greencars_type.title.first_item = 0
```

Selection Case

We are interested in cars

- with a diesel or domestic gas engine

- with a price-bracket from 20.000 EUR to 25.000 EUR

Illustration 30: Green Cars with three selectboxes - each with size = 1

The values in the boxes will be changed after sending the form:

- every item has a new number of hits

- there are some items without any hit and they won't be displayed like

  - Coupe

  - all engines without a selection (of course)

  - all price-brackets without a selection (of course)

Illustration 31: Green Cars with three selectboxes - each with size = 1

Change the Behavior

Display price-brackets even if they haven't any hit

Illustration 32: Green Cars with three selectboxes - each with size = 1

TypoScript

```
// Display price-brackets even if they haven't any hit
plugin.tx_browser_pi1 {
  views.list.1.filter.tx_greencars_pricebracket.title.wrap.item.display_without_any_hit = 1
}
```

Display price-brackets without the number of hits

Illustration 33: Green Cars with three selectboxes - each with size = 1

TypoScript

```
// Display price-brackets without the number of  hits
plugin.tx_browser_pi1 {
  views.list.1.filter.tx_greencars_pricebracket.title.wrap.item.display_hits = 0
}
```

Organiser[1] - Filter for a period (data area)

This is a typical example for a calendar.

This is the case:

- dates of events are stored in a field "datetime"
- the format of a date is a timestamp
- the table tx_org_cal contains the field datetime. The table.field-syntax is: tx_org_cal.datetime

Preconditions

We have configured a list view (and a single view of course) for the table tx_org_cal.

HTML Template

We have added the marker ###TX_ORG_CAL.DATETIME### to our HTML template into the filter section.

See section HTML Template on page 13 above.

TypoScript Snippet SELECTBOX

```
1.  plugin.tx_browser_pi1 {
2.    views {
3.      list {
4.        201 {
5.          name    = Calendar
6.          select (
7.            tx_org_cal.title,
8.            tx_org_cal.datetime
9.          )
10.          csvLinkToSingleView = tx_org_cal.title
11.          filter {
12.            tx_org_cal {
13.              datetime < plugin.tx_browser_pi1.displayList.master_templates.selectbox
14.            }
15.          }
16.        }
17.      }
18.      ...
19.    }
20. }
```

Result in the Frontend

Illustration 34: Calendar with select box

---

1   Organiser (extKey: org) is a powerfull database for organisations, companies and organisers: You can publish all data in real terms. You can design an organisation with headquarters, locations, staff, events, news and documents. Org displays data in list, single and calendar views. Org hasn't any PHP code. It is based on the TYPO3-Fronend-Engine Browser. You can control and develop org by plugins and TypoScript. Org is in development. We intend to publish org at the 1st of April, 2011.

Comment

Line 11 to 13 (see page above): we allocates a filter for tx_org_cal.datetime in form of a SELECTBOX.

The items of the SELECTBOX are in our case:

```
<option value="1283968800">1283968800 (2)</option>
<option value="1304892000">1304892000 (1)</option>
<option value="1315591200">1315591200 (3)</option>
```

This means:

- We have two dates at 1283968800, one date at ... and three dates at ...

But we don't want for every date an item in the selectbox. We want something like "Display the events of the last year, this year and the next year".

We want something like this:

```
<option value="10">2009 (0)</option>
<option value="20">2010 (2)</option>
<option value="30">2011 (4)</option>
```

We add an area out of the master template section and configure some properties.

TypoScript Snippet with Area

```
1.  plugin.tx_browser_pi1 {
2.    views {
3.      list {
4.        201 {
5.          name    = Calendar
6.          select (
7.            tx_org_cal.title,
8.            tx_org_cal.datetime
9.          )
10.         csvLinkToSingleView = tx_org_cal.title
11.         filter {
12.           tx_org_cal {
13.             datetime < plugin.tx_browser_pi1.displayList.master_templates.selectbox
14.             datetime {
15.               // Every item should be displayed even though it hasn't any hit
16.               wrap.item.display_without_any_hit = 1
17.               // Order the items by uid (and not by value)
18.               order.field = uid
19.               // Allocates the master template sample_period
20.               area < plugin.tx_browser_pi1.displayList.master_templates.areas.sample_period
21.             }
22.           }
23.         }
24.       }
25.     }
26.     ...
27.   }
28. }
```


Comment

Line 16: Every item should be displayed even though it hasn't any hit

Line 18: Order the items by uid (and not by value)

Line 20: This is the most important line. We allocate the master template sample_period.
          This template provides by default:
                    - a period of years
                    - a period from last year to next year (be aware of the relative period!)
                    - and the items are formatted as year only.

Every property is configurable.

We get, what we want:

```
<option value="10">2009 (0)</option>
<option value="20">2010 (2)</option>
<option value="30">2011 (4)</option>
```

Properties and Possibilities

Please refer to the Reference AREAS on page 32 below.

Reference

This tutorial provides only the reference for the master_template array (see below)

You find all other stuff in the manual of the Browser:

- HTML http://typo3.org/documentation/document-library/extension-manuals/browser/current/

- PDF: http://typo3.org/extensions/repository/view/browser/current/info/?tx_terfe_pi1%5BdownloadFile%5D=doc%252Fmanual.pdf

master_templates

All master_templates are local. You have to copy the template to your view and configure it there.

See the example below.

Property:	Data type:	Description:	Default:

master_templates	array	Configuration of master templates for the filter.
		
		Content Objects are
		  • CATEGORY_MENU
		  • CHECKBOX
		  • RADIOBUTTON
		  • SELECTBOX
		
		Code: ... **Example:** ... (see below)	

```
Code:
plugin.tx_browser_pi1 {
  displayList {
    master_templates {
      selectbox = SELECTBOX
      selectbox {
        ...
      }
    }
  }
}
```

```
Example:
plugin.tx_browser_pi1 {
  views {
    list {
      1 {
        select (
          fe_users.last_name,
          fe_users.first_name,
          fe_users.image,
          fe_groups.title,
          tx_org_management.title
        )
        filter {
          fe_groups {
            title < plugin.tx_browser_pi1.\
                    displayList.\
                    master_templates.selectbox
            title {
              first_item {
                stdWrap {
                  data >
                  value   = All Groups
                  lang.de = Alle Gruppen
                }
              }
              wrap = \
                <span class="selectbox">|</span>
              wrap {
                item {
                  stdWrap {
                    crop = 30 | ... | 1
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

CATEGORY_MENU

Display your filter as a HTML category menu.

The subpart ###CATEGORY_MENU### is disabled by TypoScript by default. The subpart will removed from the HTML template – you won't have any waste in your code.

If you are using the category menu please enable in your global or local configuration:

```
displayList.display.category_menu = 1
```

Property:	Data type:	Description:	Default:
CATEGORY_MENU	array	Content Object CHECKBOX **Example:** See Code of SELECTBOX below	

CHECKBOX

Display your filter as a HTML check box.

Property:	Data type:	Description:	Default:
CHECKBOX	array	Content Object CHECKBOX **Example:** See Code of SELECTBOX below	

RADIOBUTTONS

Display your filter as HTML radio buttons.

Property:	Data type:	Description:	Default:
RADIOBUTONS	array	Content Object RADIOBUTONS **Example:** See Code of SELECTBOX below	

SELECTBOX

Property:	Data type:	Description:	Default:
SELECTBOX	array	Content Object SELECTBOX **Example:** See Code below	

```
plugin.tx_browser_pi1 {
  displayList {
    master_templates {
      selectbox = SELECTBOX
      selectbox {
        // [String/optional] Name for the piVar. Name must be unique.
        nice_piVar =
        // [Boolean] Should be added a default item?
        first_item = 1
        first_item {
          // [Boolean] 0: Item without any hit will not be displayed. 1: Any item will be displayed.
          display_without_any_hit = 1
          // [Boolean] 1: Items display the number of hits. 0: Items doesn't display any number of hits.
          display_hits = 0
          display_hits {
            // [Boolean] 1: Hits are displayed behind the item. \
            //           0: Hits are displayed in front of the item.
            behindItem = 1
            stdWrap {
              noTrimWrap = | (|)|
            }
          }
          stdWrap {
            data = LLL:EXT:browser/pi1/locallang.xml:label_filter_firstitem_default
          }
          // [STRING, unique] Value attribute in the option tag (#11401)
          option_value = 0
        }
        // [Boolean] Should it be possible to select more than one item?
        multiple  = 0
        multiple {
          // [String] HTML code for multiple
          selected = multiple="multiple"
        }
        // [Integer] How many items should the select box display in the visible area?
        size = 1
        // [String] Wrap the object
        wrap = <div class="selectbox">###TITLE###|</div>
        wrap {
          // [Integer] Space characters from left margin in the HTML code
          nice_html_spaceLeft = 12
          title_stdWrap {
            // [String] Title of the object. If it shouldn't display, please delete the title array
            value = ###TABLE.FIELD###
            wrap  = <div class="selectbox_title">|</div>
          }
          // [String] Wrap the items
          object = \
            <select name="###TABLE.FIELD###" id="###ID###" size="###SIZE###"###MULTIPLE###>|</select>
          object {
            // [Integer] Space characters from left margin in the HTML code
            nice_html_spaceLeft = 14
          }
          // [String] Wrap one item
          item     = <option###CLASS######STYLE### value="###UID###"###ITEM_SELECTED###>|</option>
          item {
            // [Boolean] 0: Item without any hit will not be displayed. 1: Any item will be displayed.
            display_without_any_hit = 0
            // [Boolean] 1: Items display the number of hits.\
            //           0: Items doesn't display any number of hits.
            display_hits = 1
            display_hits {
              // [Boolean] 1: Hits are displayed behind the item. \
              //           0: Hits are displayed in front of the item.
              behindItem = 1
              stdWrap {
                noTrimWrap = | (|)|
              }
            }
```

```
                    // [Integer] Space characters from left margin in the HTML code
                    nice_html_spaceLeft = 16
                    // [String] HTML code for a selected item
                    selected = selected="selected"
                    // [String] Default HTML class
                    class =
                    // [String] Default HTML style
                    style =
                    stdWrap {
                      htmlSpecialChars = 1
                      crop            = 60 | ... | 1
                    }
                  }
                }
              order {
                // [String] value || uid
                field = value
                // [String] ASC || DESC
                orderFlag = ASC
              }
              sql {
                // [String, SQL] And where statement. Example: \
                    tt_news_cat.author.field LIKE 'A%' AND tt_news.uid < 99
                andWhere =
              }
              // [BOOLEAN] You have to allocate one of the areas of the master template.
              area = 0
            }
          }
        }
      }
```

AREAS

Areas contains master templates for area samples. Samples are

- currency
  - hard coded
- period
  - hard coded
  - relative

Areas are limited to integers and numbers.

We intend to extend it for chars and strings too[1].

You need a content object SELECTBOX, CHECKBOX or RADIOBUTTONS from above, if you like to use an area filter.

Property:	Data type:	Description:	Default:

areas	array	Configuration array for an area. You need it, if you like to filter data areas
		
		Default code:
		
		``plugin.tx_browser_pi1 {
		  displayList {
		    master_templates {
		      areas {
		        sample_currency = 1
		        sample_currency {
		          strings = 1
		          strings {
		            // Configuration for a hard coded
		            // currency area
		          }
		        }
		        sample_period = 1
		        sample_period {
		          strings = 0
		          strings {
		            // Configuration for a hard coded
		            // period
		          }
		          interval = 1
		          interval {
		            // Configuration for a relative period
		          }
		        }
		      }
		    }
		  }
		}```	

Example

Please refer to the section Organiser1 - Filter for a period (data area) on page above.

TypoScript Object Browser

Please use the TypoScript Object Browser, if you like to inspecting all properties.

Possibilities

See examples below.

---

1   We like a contract. Do you know someone for contracting?

AREA - Period interval

Period interval is a smart tool: Period interval enables you to define relative periods like

- display all seven days of the current week
- display four weeks from one week ago
- display alls months of the current year
- display all years from last year to next year

If you want to play with the examples, please change the case property like this:

```
plugin.tx_browser_pi1.displayList.master_templates.areas {
  sample_period.interval.case = week
}
```

Property:	Data type:	Description:	Default:

areas	array	Configuration array for an area. You need it, if you like to filter data areas
		
		Default code:
		
		`plugin.tx_browser_pi1 {`
		`  displayList {`
		`    master_templates {`
		`      areas {`
		`        sample_period = 1`
		`        sample_period {`
		`          interval = 1`
		`          interval {`
		`            // [BOOLEAN]: 1: dates are converted to\`
		`                        timestamp. 0: undefined`
		`            compare_wiTimeStamp = 1`
		`            // [STRING] day, week, month, year`
		`            case = year`
		`            day {`
		`              // Configuration for days`
		`            }`
		`            week {`
		`              // Configuration for weeks`
		`            }`
		`            month {`
		`              // Configuration for months`
		`            }`
		`            year {`
		`              // Configuration for years`
		`            }`
		`          }`
		`        }`
		`      }`
		`    }`
		`  }`
		`}`	

Day

See year below.

Week

See year below.

Month

See year below.

Year

Every period has the properties:

- start_period

- times_stdWrap

- value_stdWrap

Please inspect the TypoScript snippet below too become familiar with the properties and possibilities.

```
year {
  start_period {
    // [BOOLEAN] 1: Value is handled after stdWrap with php strtotime. 0: stdWrap only.
    use_php_strtotime = 1
    stdWrap {
      // Examples with strtotime: -1 year, 2010, now, next year
      value = -1 year
    }
  }
  times_stdWrap {
    // [INTEGER] Display the period n times
    value = 3
  }
  value_stdWrap {
    // [STRING] Leave it empty! It will be handled while runtime.
    value =
    // [STRING] See php method strftime
    strftime = %Y
  }
}
```

Example

Please refer to the section Organiser1 - Filter for a period (data area) on page above.

AREA - Period hard coded

It is possible to "hard code" a period too. See the example in

```
plugin.tx_browser_pi1.displayList.master_templates.areas {
  sample_period.string {
    ...
  }
}
```

AREA - Currency hard coded

It is possible to "hard code" an area of numbers too. See the example in

```
plugin.tx_browser_pi1.displayList.master_templates.areas {
  sample_curreny.string {
    ...
  }
}
```

What's new in Version 3.6?

Filter for areas

You can filter your data by areas. Example for areas are

- 2010 to 2012
- 10.000 EUR to 20.000 EUR

You can generate filter for periods relative like:

- all days of the current week
- -2 weeks from now to +2 weeks from now

Filter for local table

You can filter data from the local table. Before version 3.6.0 it was possible to filter data of foreign tables only.

Category Menu

You can filter your data by a category menu. The difference between a category menu and the available filters check box, radio buttons and select box is, that the first is a list of links and all others are a part of an HTML form.

You can use areas for a category menu in the same way like for filters.

Configurable: Search respects Filters

It is configurable, if the search respect filters. It is possible to differentiate between every filter.

The new TypoScript property is

```
plugin.tx_browser_pi1.displayList.display.searchform.respect_filters
```

Helpful suggestions

Forum

The Browser has its own forum. We speak English and German.

You are welcome to post any question, bug or snippet code at

- TYPO3-Frontend-Engine Browser: Forum
  http://typo3-browser-forum.de/

Illustration 35: The TYPO3-Frontend-Engine Browser Forum

Credits

dorn & malag

Thanks to

> dorn & malag
> agentur fürs netz
> http://www.dorn-malag.de/

who sponsored a part of the features CHECKBOX and RADIOBUTTONS.

Illustration 36: http://www.dorn-malag.de/

LEG Thüringen

Thanks to

> LEG Thüringen
> http://leg-thueringen.de/

who sponsored a part of the feature to filter data in local tables.

Illustration 37: http://leg-thueringen.de/

To-Do List

Nothing to do.

Change Log

3.6.0:   **New Features**
- #11580: Configurable: Should the search respect filters?
- #11523: Hide the update report for public
- #11402: Filter for areas
- #11401: Filter for local table
- #8807: Category Menu

3.5.0:   **New Features**
- #8335: Filter with multiple select
- #8337: Filter with checkbox and radiobuttons
- #9408: Display the number of hits in the filter items

3.4.0:   **Update**
Conditions for filtering

3.0.4:   **Update**
Filters are documented

3.0.3:   **Update**
Plugin has new properties for the search form.

3.0.1:   **Publishing**

2.6.1:   **Initial release**

Illustration Index