Addresses

Extension Key: am_project

Copyright 2007-2008, Manuel Thaler, <m.thaler@meincms.ch>

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

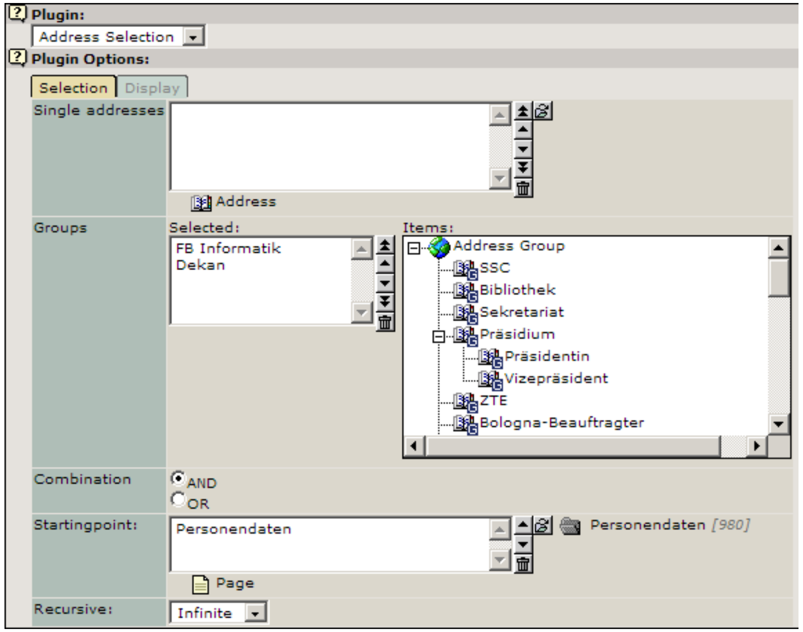
Introduction

What does it do?

This extension helps you managing and administrating projects and customers.

Screenshots

This is the plugin with the tree selector.



Plugin: Address Selection

Plugin Options:

Selection | Display

Single addresses: Address

Groups: Selected: FB Informatik, Dekan

Items: Address Group, SSC, Bibliothek, Sekretariat, Präsidium, Präsidentin, Vizepräsident, ZTE, Bologna-Beauftragter

Combination: ● AND ○ OR

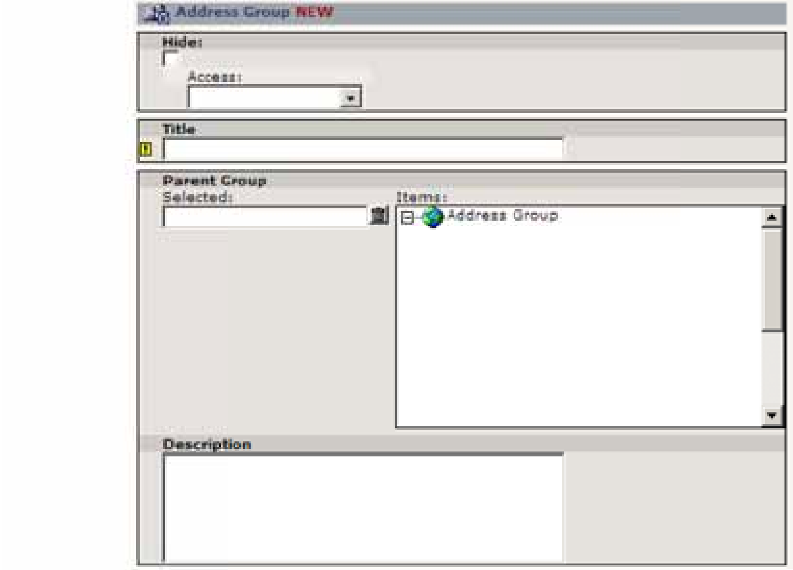
Startingpoint: Personendaten — Page — Personendaten [980]

Recursive: Infinite

Users manual

Creating a Group

At first you need to create some groups for your address records. To do this navigate to a sysfolder and switch to the list view. Click “Create new record” at the bottom, now click Address Group.



Address Group NEW

Hide: □

Access:

Title

Parent Group

Selected:

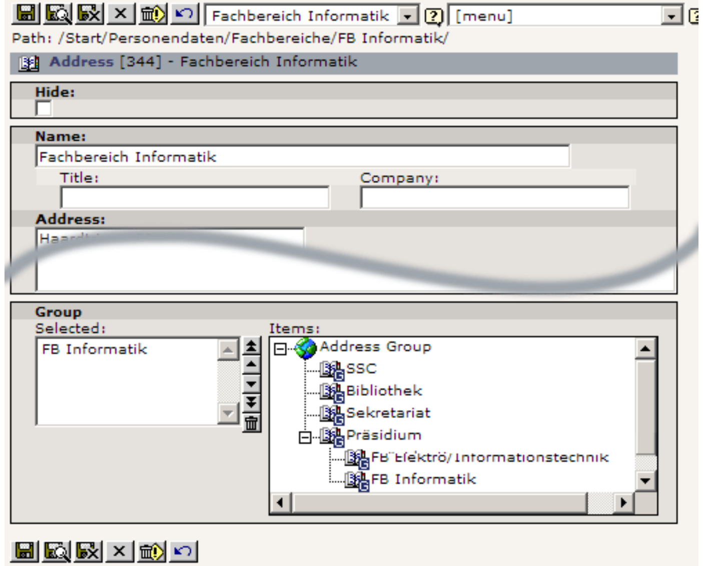
Items: Address Group

Description

In that form the title is the only required field, if you have created some groups already you will see them in the tree view and can select one as the parent group for the group you are about to create. At the bottom you can add a small note or description to the group.

Assigning groups to an address record

Adding groups to an address is as easy as adding a parent group to another group. Open the address record you want to assign a group to or create a new address record, go to the bottom of the form and add one or more groups from the right box by clicking on the group.



Path: /Start/Personendaten/Fachbereiche/FB Informatik/

Address [344] - Fachbereich Informatik

Hide: □

Name: Fachbereich Informatik

Title:

Company:

Address:

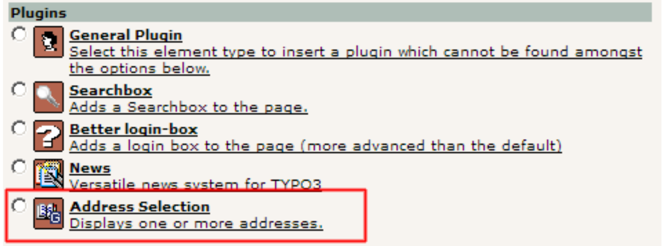
Group

Selected: FB Informatik

When adding groups to an address there's one point to be aware of: The top most group you assign to an address can be put out in the front end later, so you might want to choose that carefully or switch positions of the selected groups.

Creating a Plugin Content Element and adding addresses to it

Go to the page view and navigate to the page where you want to insert one or more addresses. Create a new content element and in the "new content element wizard" scroll down to the plugins section and select "Address Selection"



Plugins

○ **General Plugin**
Select this element type to insert a plugin which cannot be found amongst the options below.

○ **Searchbox**
Adds a Searchbox to the page.

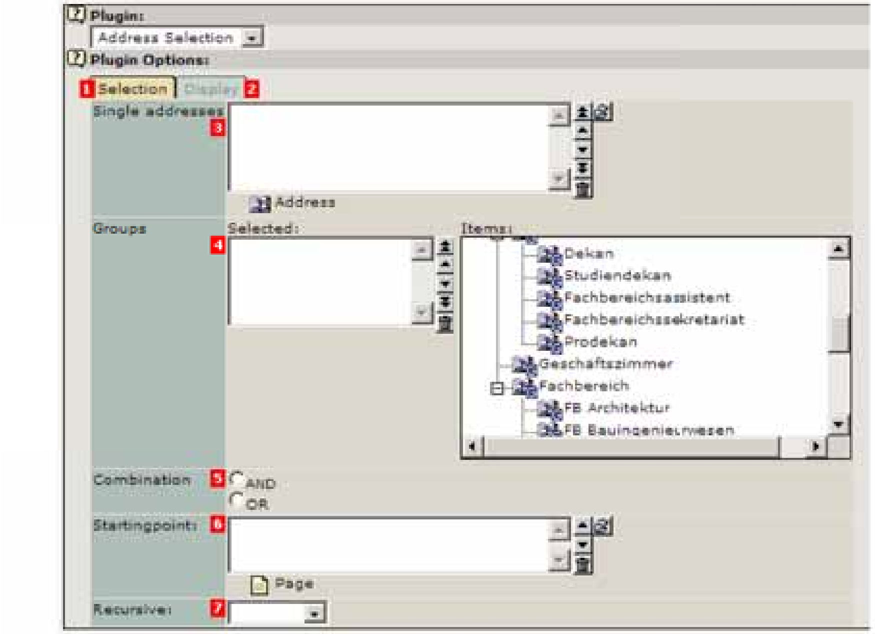
○ **Better login-box**
Adds a login box to the page (more advanced than the default)

○ **News**
Versatile news system for TYPO3

○ **Address Selection**
Displays one or more addresses.

Now on the first tab [1] you can add address records to the plugin, the second tab [2] is for selecting a template for how the address should look like, this needs to be configured by your administrator. If there're no templates on the second tab, just save the content element once and after saving it they should apear.

In the Single addresses field [3] you can select addresses by using the known element browser. Even if the addresses selected here would be selected via groups, too they do not appear twice in the front end.



Plugin: Address Selection

Plugin Options:

In the groups field [4] you can select groups from the tree view. These groups get combined either by AND or by OR [5]. That means you can select addresses which are in all of the given groups (AND) or they are required to be in at least one of the given groups (OR).

The starting point does not necessarily need to be set. The extension can be configured to work with a default setting which needs to be adjusted by your administrator. If you set a startingpoint here, it will override that default setting that might be set. The recursive field [7] sets how deep in the page tree beginning from the startingpoint the system should look for addresses with the given groups.

Adminstration

There's not much to say about administration here, just install the extension. After installation you need to configure the extension. For that, please consult the next section.

Updating from tt_address versions before 2.0.0

If you're updating from a tt_address version < 2.0.0 and the addressgroups extension there's an update script for you!

Go to the extension manager, select tt_address and in the upper right corner you'll find a drop down menu with an element called "UPDATE!". The update script will split names from the old combined name field to first and last name. Although there's also a new middle name field it is not used by the update script. Names are split by comma (expecting the part in front of the comma to be the last name and the part after the comma to be the first name) or by space (expecting the first word to be the first name and the part after the first space to be the last name even if there're more space). Records which couldn't be updated automatically will be highlighted in the update log afterwards, edit them by hand.

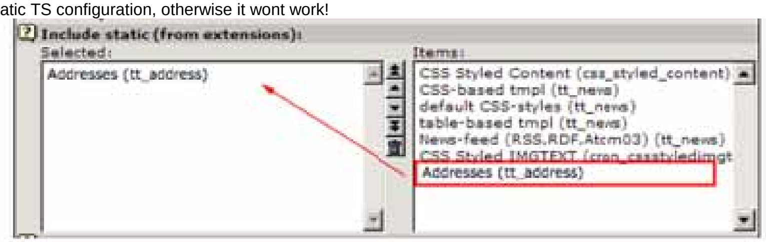
If you have installed EXT:addressgroups this will also be taken into account. The update script copies the groups to the new tt_address native table and rebuilds the relations. Addressgroups plugin content elements will also be updated to tt_address plugin content elements.

After that you need to update your HTML templates by hand, the ###NAME### marker from addressgroups is not available in tt_address, instead you need to use the respective markers ###FIRSTNAME##, ###MIDDLENAME###, and ###LASTNAME###. Also don't forget to update the TS from plugin.tx_addressgroups_pi1 to plugin.tx_ttaddress_pi1!

If you're using other extensions which extend tt_address there's one more thing which needs your attention:
After you updated your records to use the split name fields you can disable the old field from the extension manager. Althoug you disabled the old combined name field from displaying in the back end tt_address takes care of writing changes from the split name fields back to the combined name field. From the extension manager you can configure the format to use or which fits your previous record naming. Thus other extensions can still use the old – now internal – combined name field.

Configuration

When starting to configure the extension you should create a separate TS extension template for it where you do all the configuration for this extension. That extension template can then get included in the root template. In the extension template add the static TS configuration, otherwise it wont work!



Selected: Addresses (tt_address)

Items:
- CSS Styled Content (css_styled_content)
- CSS-based tmpl (tt_news)
- default CSS-styles (tt_news)
- table-based tmpl (tt_news)
- News-feed (RSS,RDF,Atom03) (tt_news)
- CSS Styled IMGTEXT (cron_cssstyledimgt
- Addresses (tt_address)

Reference

Property:	Data type:	Description:	Default:
templatePath	string	Defines the path where the templates are located. Put templates in that folder and they will be listed automatically in the plugin. You can also add a small graphic showing how the address will be displayed. That graphic need to have the same name as the template but must be a gif. This basically works like the template selector from MTB2.	fileadmin/templates/
defaultTemplateFileName	string	Defines the default template file.	addressgroups_default.htm
pidList	int	A comma separated list of integers representing page ids where to get the address records from	0
recursive	int	Defines how many levels to search for tt_address records from the given pages in pidList.	0
wrap	string	wraps the whole output	
singleSelection	string	Comma separated list of tt_address record uids, will be overriden with flexform data	
groupSelection	string	Comma separated list of tt_address group record uids, will be overriden with flexform data	
combination	int	0 = AND, 1 = OR	0
sortByColumn	string	Defines by which tt_address column you want to sort, if an invalid column is given it's set to 'name' Valid columns for sorting: uid, pid, tstamp, name, title, email, phone, mobile, www, address, company, ciy, zip, country, image, fax, description	name
sortOrder	string	Defines the sorting order, DESC for descending and ASC for ascending. Any other (invalid) value is set to ASC.	ASC
templates.[TEMPLATE_NAME]	wrap	wraps a single address	

templates.[TEMPLATE_NAME].[FIELD_NAME]	stdWrap	The configurations for the different templates goes here. Let's have a look at an example:
		Let's assume you have three different html templates:
		* template_1.htm
		* other_template.htm
		* different_template.htm
		TEMPLATENAME is the file name without the extension. Now you can configure each of these templates:
		
		Example:
		plugin.tx_ttaddress_pi1 {
		  templates.template_1 {
		    email.wrap = E-Mail: |<br />
		    email.required = 1
		    ...
		  }
		
		  templates.other_template {
		    ...
		  }
		
		  templates.different_template {
		    ...
		  }
		}
		
		each standard tt_address field can be configured with stdWrap properties. Like wrap and required which will propably be the most important and most used ones.
		
		Here's the list of default fields you can use inside each template configuration:
		first_name, middle_name, last_name, title, email, phone, mobile, www,, address, company, city, zip, region, country, image, fax, description, mainGroup
		
		You can control max width and hight of an image with image.file.maxW and image.file.maxH
		
		An example template can be found in the res folder in the extension.	

templates.[TEMPLATE_NAME.][subparts]		Each template configuration can have subparts.
		You can define as many subparts as like, stdWrap properties will apply to them. If you have a subpart configuration
		
		  templates.other_template {
		    ...
		    subparts {
		      xyz {
		
		      }
		      abc {
		
		      }
		    }
		  }
		you can use these subparts in your HTML template like
		<!-- ###SUBPART_XYZ### begin -->
		...
		<!-- ###SUBPART_XYZ### end -->
		<!-- ###SUBPART_ABC### begin -->
		...
		<!-- ###SUBPART_ABC### end -->
		
		Subparts have a special condition property: hasOneOf, with this property you define that an address record must have at least one of the defined name fields before the whole subpart is shown:
		subparts.abc.hasOneOf = city, zip, country	

[tsref:plugin.tx_ttaddress_pi1]

Known problems

None so far

To-Do list

Nothing yet, please add your ideas for improvements to the bugtracker.