FirstSpirit™
Unlock Your Content

FirstSpirit™ Release Notes
FirstSpirit™ Version 5.1

Version	**1.03**
Status	**RELEASED**
Date	**2015-02-03**
Department	FS-Core
Copyright	2015 e-Spirit AG
File name	RELN_EN_FirstSpirit_Releasenotes

e-Spirit AG

Stockholmer Allee 24
44269 Dortmund | Germany

T +49 231 . 477 77-0
F +49 231 . 477 77-499

info@e-Spirit.com
www.e-Spirit.com

e-Spirit

Table of content

1 Introduction

This document describes the newly implemented functions in FirstSpirit V5.1. As a prerequisite, the reader must be familiar with FirstSpirit™ and must have sufficient technical background knowledge. In particular, in-depth knowledge of the relevant fields (template development, administration) is required to understand chapters 7 to 9. Chapter 2 outlines the highlights of this version.

The FirstSpirit applications have been renamed in FirstSpirit version 5.1. These applications are now known as ContentCreator, SiteArchitect, ServerManager, and ServerMonitoring (see section 5.1, starting on page 15 for more information).

This document is provided for information purposes only. e-Spirit may change the contents hereof without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. e-Spirit specifically disclaims any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. The technologies, functionality, services, and processes described herein are subject to change without notice.

2 Highlights of FirstSpirit 5.1

2.1 Renaming FirstSpirit applications

The Global Experience ideas presented in the FirstSpirit Roadmap and other areas resulted in the renaming of FirstSpirit applications in version 5.1.

The following terms are now being used:

- FirstSpirit ContentCreator
- FirstSpirit SiteArchitect
- FirstSpirit ServerManager
- FirstSpirit ServerMonitoring

From the point of view of usability, **ContentCreator** offers an optimized editing environment for efficient content management and so is aimed directly at editors. The relevant button, which can be used to start the individual FirstSpirit applications, is located in a prominent position on the FirstSpirit start page. This emphasizes, in a visual way, the importance of this application for the primary FirstSpirit user group (= editors).

SiteArchitect is generally used by a smaller number of users. It has been designed for the configuration of projects and the development of templates and so is primarily aimed at template and project developers. However, it can also be used to carry out more complex editing tasks ("power user").

Also refer to section 5.1, page 15 for more information.

2.2 Design refresh in SiteArchitect

FirstSpirit version 5.1 has also added new features to **SiteArchitect**, thus providing an even better user experience. The SiteArchitect software interface not only has a fresh, more streamlined look and feel, but also features improved performance thanks to refactoring "under the hood."


Welcome, Martina Marketing!
FirstSpirit™
Unlock Your Content
Project History
Last Edited
Show Report
My Actions
Neue Projektnews
Mitarbeiter hinzufügen
Zeitsprung vorbereiten
Project Dashboard
Direct access to useful and often-used project information and functions
Customizable per user and project
products (Current data records)
Filter [All Records]
Sort Order
Optimized Dataset Handling
A better view of data sources and dataset forms
Easily filter, search, and sort
produkte_1
Content
Metadata
Site structure (root)
products
produkte_1
General
File name
products_overview
Keywords
Options
Show in page group
Design Refresh
Intuitive interaction due to improved user interface layout
Shorter click paths for faster results

2.3 Multi-perspective preview: content in all dimensions

The ubiquity of Internet-capable smartphones and tablets has caused many companies to change their thinking in recent years. To ensure the successful implementation of web projects, it is no longer enough to have high-quality content that is precisely tailored to the respective target group; rather, it is becoming increasingly important to ensure that the content is properly formatted for a variety of output devices with different resolutions and display sizes. In light of this, FirstSpirit is putting its faith in sustainable website concepts such as "responsive design" and "mobile first" so that the quality of all web content remains consistent in the long term.

Version 5.1 offers a convenient way to check the display of content and navigation using different display sizes in the ContentCreator preview and to adapt it to suit the output device. The aim here is to minimize the amount of effort involved in developing responsive layouts and maintaining the pages implemented on this basis.

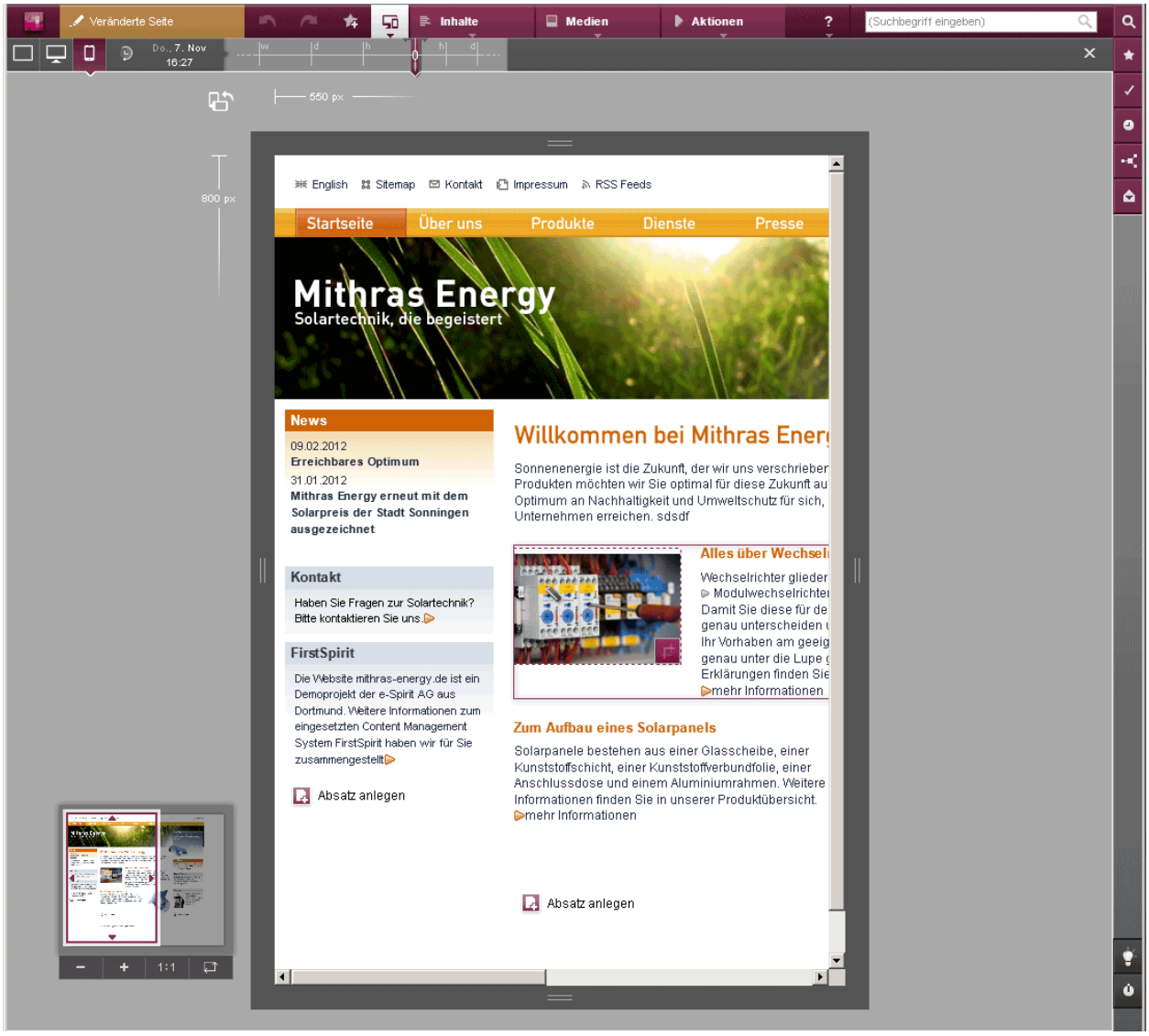


Figure 2-1: Mobile content preview in ContentCreator

In addition to the display size of the different output devices, the editor may also be interested in the chronological development of a page or (for example) what the page looks like to user-specific roles. To cater specifically to scenarios that involve controlling marketing campaigns or the activation of time-limited offers in the online shop, FirstSpirit 5.1 has a convenient preview feature that shows what each page will look like at a given time. Likewise, the past development of the page can also be tracked. For this purpose, a time bar function has been implemented. The page's development over time is displayed continuously in the preview and you can use a slider to select the required point in time.

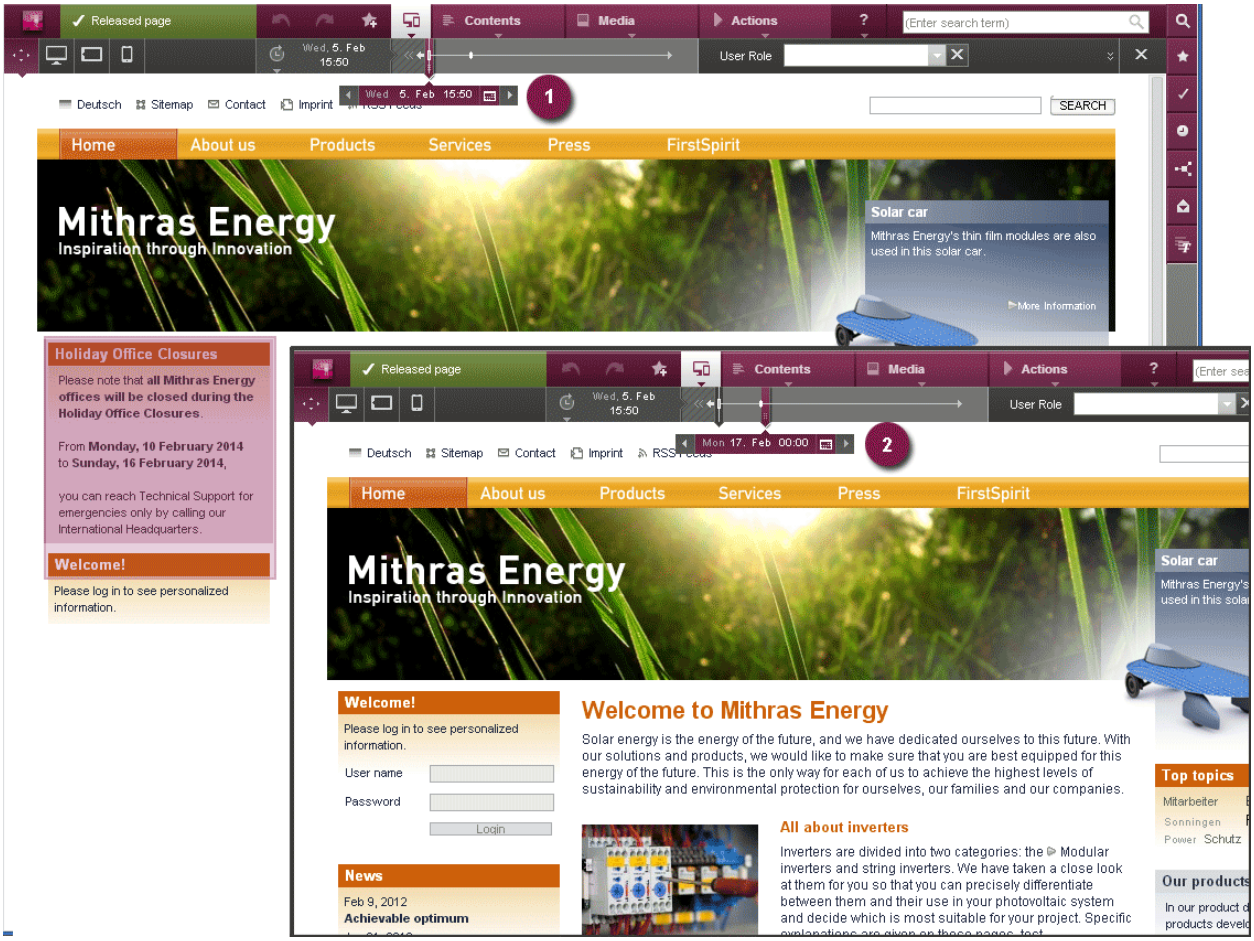


Figure 2-2: Time-dependent change to the content of a project

Also refer to section 6.1.6, page 31 and section 7.4.5, page 103 for more information.

2.4 Developer experience: faster and error-free development with FirstSpirit

FirstSpirit version 5.1 is the start of a development phase focused more on improving the development processes that take place when executing complex FirstSpirit projects.

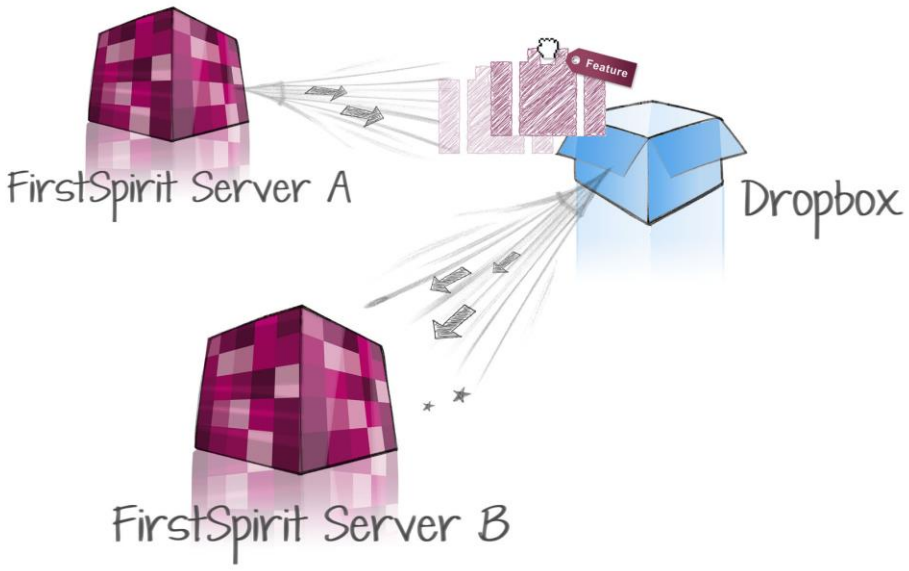
Enhancements in version 5.1 related to this include the FirstSpirit Debugger as well as Template Highlighting, which gives the developer the ability to display the tag structure of the current page's HTML code in the integrated preview and from there to open the related FirstSpirit templates in the workspace. The debugger can be used to execution paths of the template generation can be completed in detail based on the particular HTML page currently displayed in the preview. This also applies to all dependent templates (including templates for sections, tables, formats, or links).

Also refer to section 7.1, page 55 for more information.

The new "External synchronization" function, on the other hand, can be used to export partial structures from FirstSpirit (templates, store subtrees, etc.) to a file system structure with a folder hierarchy – in a format that is "legible to humans" and "can be interpreted by IDEs": There (that is, outside of FirstSpirit), the exported files can be edited and then resynchronized with FirstSpirit. This enables the user to synchronize changes in the file system with changes in the FirstSpirit project using a process that is largely automatic. As a result, any changes made to a template in the IDE are immediately reflected in the FirstSpirit project belonging to the developer.

Also refer to section 7.2, page 58 for more information.

Starting with FirstSpirit 5.1, more support is available for development within decentralized teams through the use of the license-based "Content Transport" module. This module allows for the creation of an infrastructure with which a user interface can be used to replace FirstSpirit content via external storage media, including across servers, such as via Dropbox:



The replacement or exchange of FirstSpirit content (such as templates that are to be edited by multiple developers as part of DQP scenarios (development, QA testing, product implementation) can now also be automated using FirstSpirit schedule management.

Also refer to section 9.2, page 129 for more information.

2.5 Application integration enhancements

The majority of application integration options for both SiteArchitect and ContentCreator have been enhanced in FirstSpirit version 5.1. In ContentCreator, for example, this includes the ability to drag report data not only to certain areas of the preview, but into open forms as well. Reports in SiteArchitect can now be integrated in the Organize area in the left-hand client column.

Also refer to section 6.2.3, page 41, section 7.3.3, page 96 and section 7.5.2, page 107 for more information.

3 System requirements

For detailed information regarding the system requirements of FirstSpirit version 5.1, see the *FirstSpirit technical data sheet* for version 5.1.

Note: Use of Oracle JDK 1.7.0_51 will cause security dialogs to appear when starting FirstSpirit SiteArchitect or ServerManager. These cannot be resolved on e-Spirit's end. For more information, see:
http://docs.oracle.com/javase/7/docs/technotes/guides/jweb/security/securityDialogs.html.

4 Switching from older FirstSpirit versions

For information on new installations and **upgrading** to FirstSpirit version 5.1, please refer to the *FirstSpirit installation instructions* for version 5.1.

FirstSpirit version 5.0 introduced a **new ID format for datasets** so that they could be more easily identified. The IDs are now created based on the UUID standard[1]. The conversion will take place automatically for existing projects when the datasets are edited in the FirstSpirit. project. This means that newly added or saved datasets will automatically receive a GID ("Global ID") in FirstSpirit versions 5.0 and higher.

In FirstSpirit version 5.1, the default generation logic for GIDS will change to randomly generated, globally unique IDs, such as the following:

```
30b88864-8c4b-41f9-923a-cfe977d489e0
```

Changing the default generation logic may lead to mixed states in projects, since when projects were migrated to FirstSpirit version 5.0, the existing datasets in the project initially had a GID (in the legacy GID format). The GIDs of newly added datasets, however, have now been generated in the new random GID format in FirstSpirit version 5.0 and higher.

[1] http://docs.oracle.com/javase/1.5.0/docs/api/java/util/UUID.html

The change in internal ID format may require migration to be done manually in projects in some rare instances. An exception to this are projects that were created in FirstSpirit versions prior to 5.0 and that use the license-dependent **Content Transport** function to transport database content. In these cases, manual adaptation of the GIDs (and resolving of mixed states in the source project (see above)) is recommended. During migration, make sure that the GIDs of the datasets from the source project can be assigned to the corresponding datasets from the target project. This assignment largely depends on the technical requirements of the affected projects. It is therefore not possible to provide a pre-developed migration solution for these situations.

The relevant methods for manually adapting or changing the GIDs are available via the `GidAgent` interface, which is part of the FirstSpirit Developer API.

Note: Making manual GID generation changes should not be taken lightly and should be done with care. The GIDs are used internally by FirstSpirit to reference the datasets, for instance in dataset-based input components (FS_LIST type database, FS_DATASET). These considerations should be taken into account particularly in the case of a project-specific implementation in order to prevent errors from occurring in the project as a result of the migration.

Changes have also been made to **license management** in version 5.1. See paragraphs on "License **management**" and "Workflows" in section 10.1 for more information.

The use of "old" input components, "old" link template types or API calls, for example, can lead to **deprecation warnings.** Log files should be examined after these warnings and the cause(s) eliminated.

Moreover, it is recommended for all FirstSpirit server updates to newly create all self-produced **modules** vis-a-vis the new FirstSpirit version. In addition to this, all the installed modules delivered by e-Spirit and all web applications should also always be updated!

Due to improvements to the internal storage format in version 5.1, some features such as **downgrades** to older versions of the product are not supported.

5 New/changed functions for all user groups

5.1 Revised FirstSpirit start page

The appearance of the start page that opens when users log into FirstSpirit has been modernized.

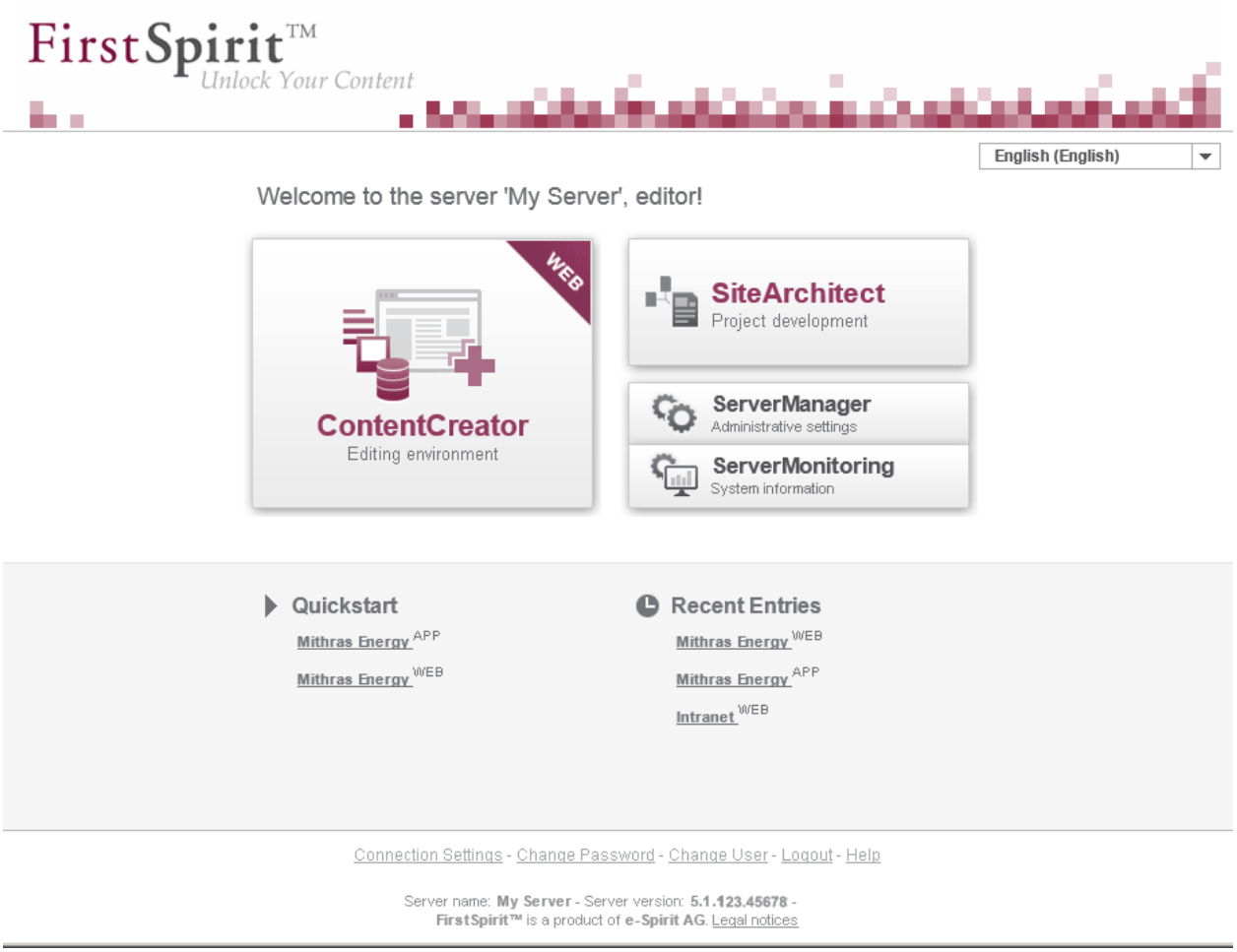


Figure 5-1: FirstSpirit start page in version 5.1

5.1.1 Application area

The Application area groups together the previous areas for clients and administration. The following applications are available:

- **ContentCreator** (previously "WebClient") is the main work environment for all editorial work carried out within a project.
- **SiteArchitect** (previously "JavaClient") is used for project development and project administration.
- **ServerManager** (previously "Server and Project Configuration") is used to perform all general administrative tasks in FirstSpirit.
- **ServerMonitoring** is used as before to monitor the FirstSpirit Server.

ServerManager and ServerMonitoring can only be started by server and project administrators. Depending on the project configuration, ContentCreator may be deactivated.

5.1.2 Project area

The Project area contains all entries which are linked directly to a project. The displayed projects can be started by clicking straight through from the start page.

- Entries displayed under **Quick-start** can be configured in the ServerManager. Each project displayed here is automatically started with the linked application. Projects linked to ContentCreator are shown with *WEB* after their name; those linked to SiteArchitect are shown with *APP*.
- Projects recently edited by the user are displayed under **Recent Entries**. Each project is automatically started with the relevant application.

5.1.3 User area

The User area contains the user settings for the current user as before.

5.1.4 Improved support for right-to-left script (RTL)

The text input components

- Single-line text (CMS_INPUT_TEXT)
- Multi-line text (CMS_INPUT_TEXTAREA)
- Rich text editor (including "DOM Editor", CMS_INPUT_DOM)
- Tables (including "DOM table", CMS_INPUT_DOMTABLE)

have been optimized for use with languages that are written from right to left ("right-to-left", abbreviated as "RTL"), such as Arabic or Hebrew.

This also affects placing and moving the cursor in the component by using the keyboard, using the mouse pointer or keyboard to highlight text, entering letters, numbers and punctuation, mixing RTL and LTR text, and entering text from other sources (such as MS Word).

6 New/changed functions for editors

6.1 New/changed functions in ContentCreator

6.1.1 Extended drag-and-drop

A major feature included in the redevelopment and enhancement of ContentCreator 5 is support for the intuitive, time-saving drag-and-drop control option. Version 5.1 now provides a basis for moving and/or referencing a wide range of data (types) in ContentCreator using drag-and-drop. This now makes it possible, for example, to drag report data not only to certain areas of the preview, but into open forms as well. The following use cases are possible:

Media (e.g. images) can be dragged

- from the Report area (e.g. from the search) to
    - the input element for reference selection (FS_REFERENCE) in the edit dialog (cf. Figure 6-1)
    - the input element for image galleries (images only; FS_LIST, type: DATABASE, media mode) in the edit dialog
    - a different medium in the preview

  and referenced in the new location.
- can be dragged from the desktop to
    - the input element for reference selection (FS_REFERENCE) in the edit dialog (cf. Figure 6-1)
    - the input element for image galleries (images only; FS_LIST, type: DATABASE, media mode) in the edit dialog
    - a medium in the preview

  in order to create links.

If the media do not come from the project, the project developer may be able to upload them using the drag-and-drop process, depending on the configuration.

Pages can be dragged

- from the Report area (e.g. from the search) to
    - the input element for reference selection (FS_REFERENCE) in the edit dialog (cf. Figure 6-1)

- the rich text editor (CMS_INPUT_DOM) or the rich text editor for tables (CMS_INPUT_DOMTABLE) for creation (for more information, see section 6.1.5 page 27)

in order to create links.

Datasets can be dragged

- from the Report area (e.g. from the search) to
  - the input element for dataset selection (FS_DATASET)
  - the input element for creating dataset lists (FS_LIST, type: DATABASE)

and referenced in the new location.

Possible drop zones are highlighted in color to make operation more intuitive. Drag actions involving data types which do not match the drop zone are rejected.

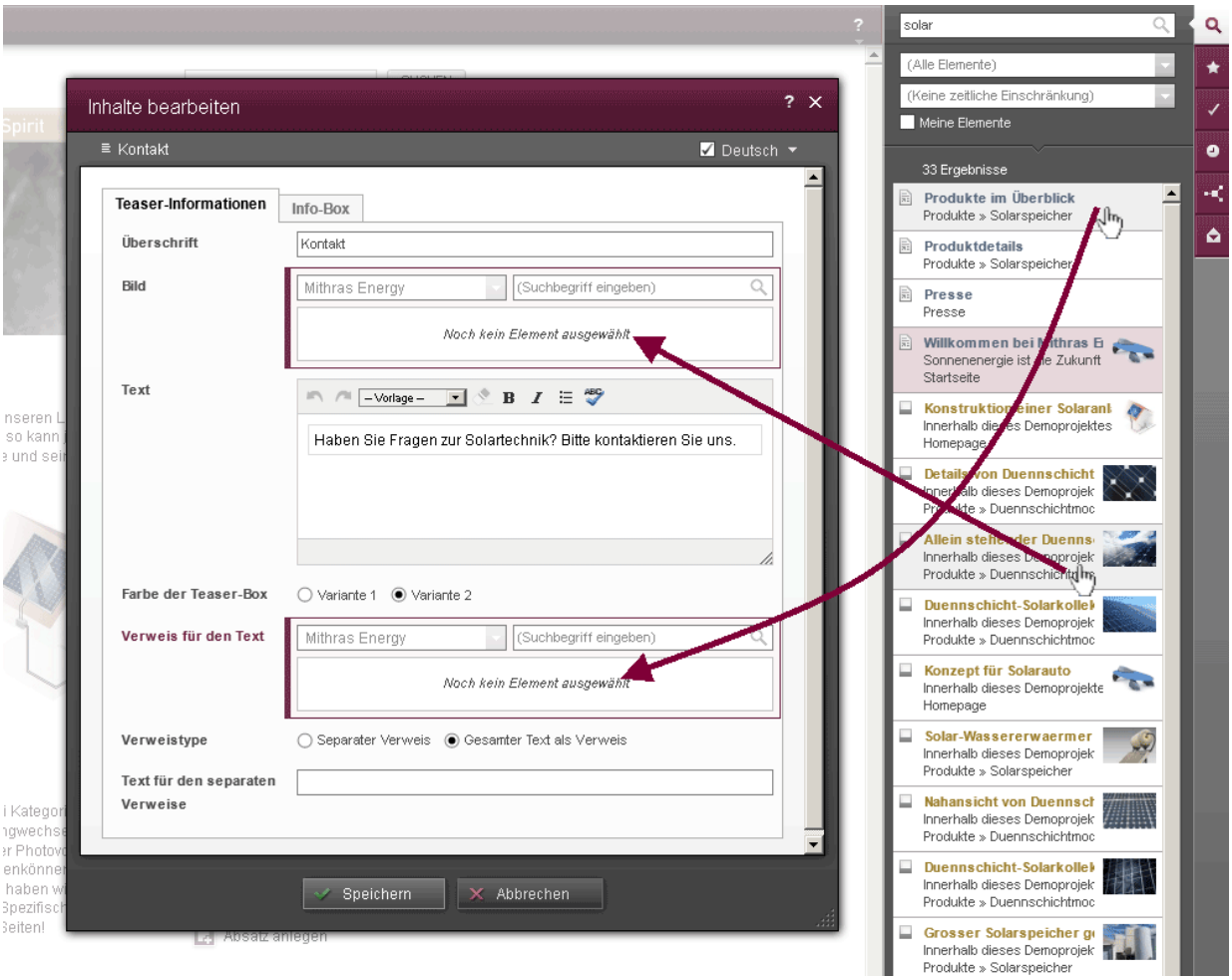


Figure 6-1: Drag-and-drop from report to input element

If an editing window is open, no information in tooltips will be shown for report entries and no functions can be executed on these entries (for example jumping to the object in the project).

- To move menu items on the preview page using drag-and-drop, see section 6.1.3 page 21.
- To create links using drag-and-drop, see section 6.1.5 page 27.

6.1.2 New input element for mouse-sensitive images ("image map")

The new input element for mouse-sensitive images can be used to embed links at various locations in a selected background image. First, a rectangular frame is created for each link. The size and shape of the frame can be adjusted to suit the desired location in the background image. The link can then be embedded in this frame (or "mouse-sensitive area").

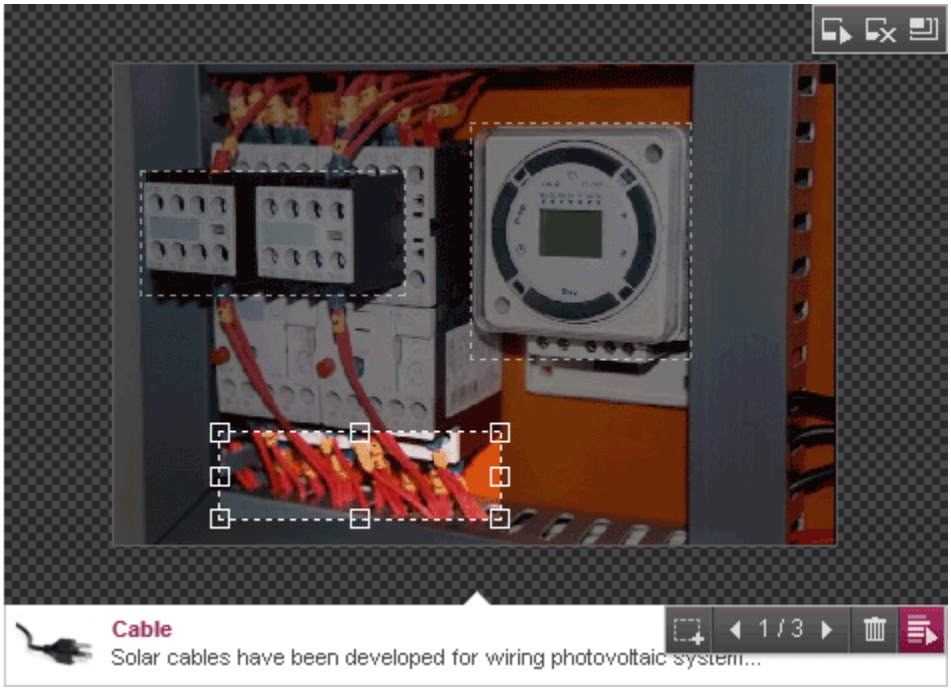


Figure 6-2: Input element for mouse-sensitive images

For detailed information, see *FirstSpirit documentation for ContentCreator*, "Edit preview page"/"Input elements"/"Mouse-sensitive images".

6.1.3 Moving menu items on the preview page

Previously, pages or menu items in ContentCreator could be moved via a dialog which visualizes the project navigation ("Contents" menu/"Edit navigation").

With FirstSpirit version 5.1, it is now possible to move pages or menu items directly on the preview page.

If the template developer has configured the relevant setting, corresponding icons are displayed on the navigation elements/menu entries when the user hovers the mouse over them in the preview. These icons can be used to move the menu items within the preview:

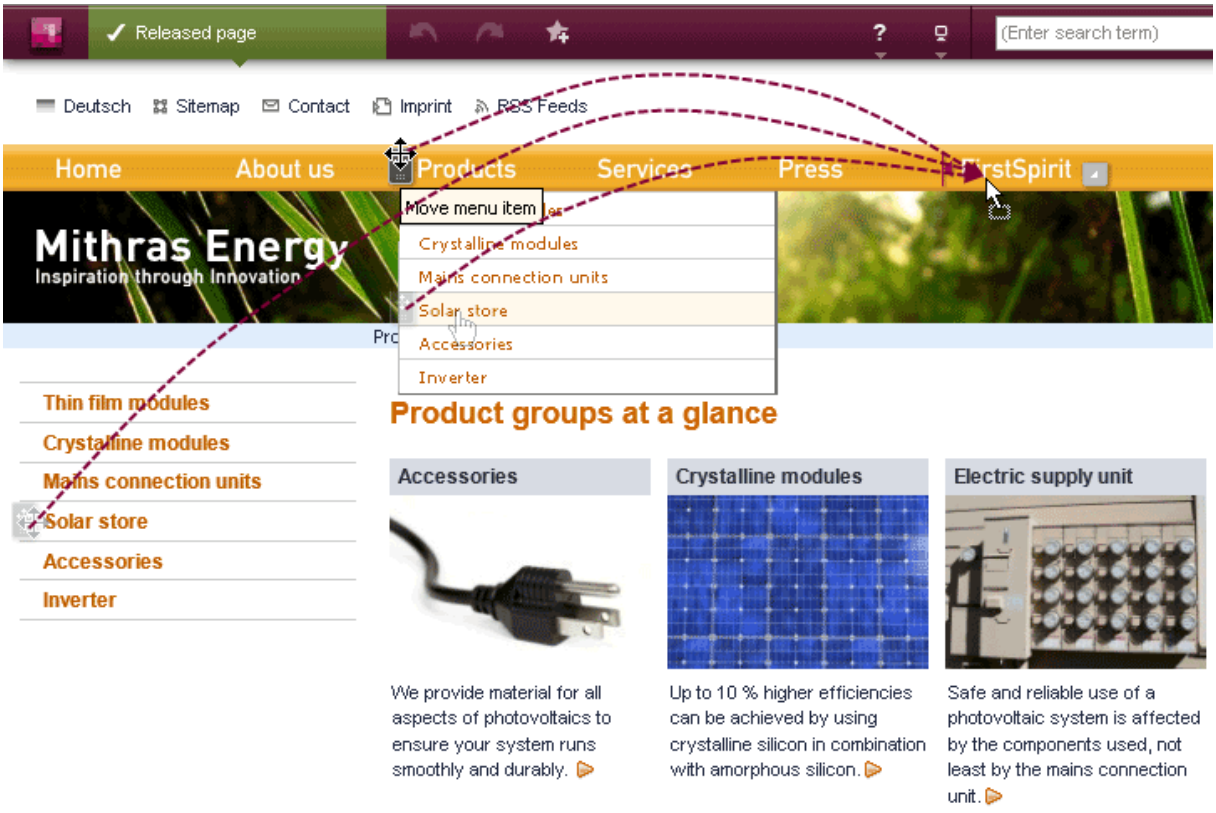


Figure 6-3: Moving menu items directly in the preview

Menu items can be moved

- in front of or behind another menu item on the same menu level
- up or down one menu level.

 Move menu item: Menu items/navigation elements with this icon can be moved. To do this, click on the icon and move the item to the desired location while holding down the mouse button.

When you hover the mouse pointer over other menu items on the page, the following icons indicate the potential drop zones:

 Insert menu item to the left

 Insert menu item to the right

 Insert menu item above

 Insert menu item below

 If some of the lower menu levels are not visible on the current page, hold the mouse pointer over this icon to show the other menu levels.

 The lower menu levels are shown as drop zones. If there are no lower menu levels available, the message "No entries found" is displayed.

If the menu item is moved to a different menu level, a confirmation prompt is displayed: "Do you really want to move the menu item?". If the user clicks "Cancel", the process is canceled and the menu item is not moved.

If the menu item is moved within the same level (changing the order of the menu items), the confirmation prompt is not displayed.

6.1.4 Personalized search results

The search function is used as a navigation and selection option in many areas of ContentCreator. Various improvements have been implemented in the new version to allow the editor to find the element they are searching for more quickly. For example, elements recently selected or edited by the editor and those edited recently in the project in general are displayed at the top of the search results. Furthermore, clicking in a search field supplies an initial list of appropriate elements (relevant to the particular input element and/or the configuration by the template developer).

Examples:

- When **creating new pages** using the "Create new page" function in the "Contents" menu in the menu bar
    - the templates most recently selected by the editor are displayed at the top when the "Empty page" option is selected
    - the pages most recently created or edited by the editor or by other users are displayed at the top when the "Apply layout" and "Copy content" options are selected.

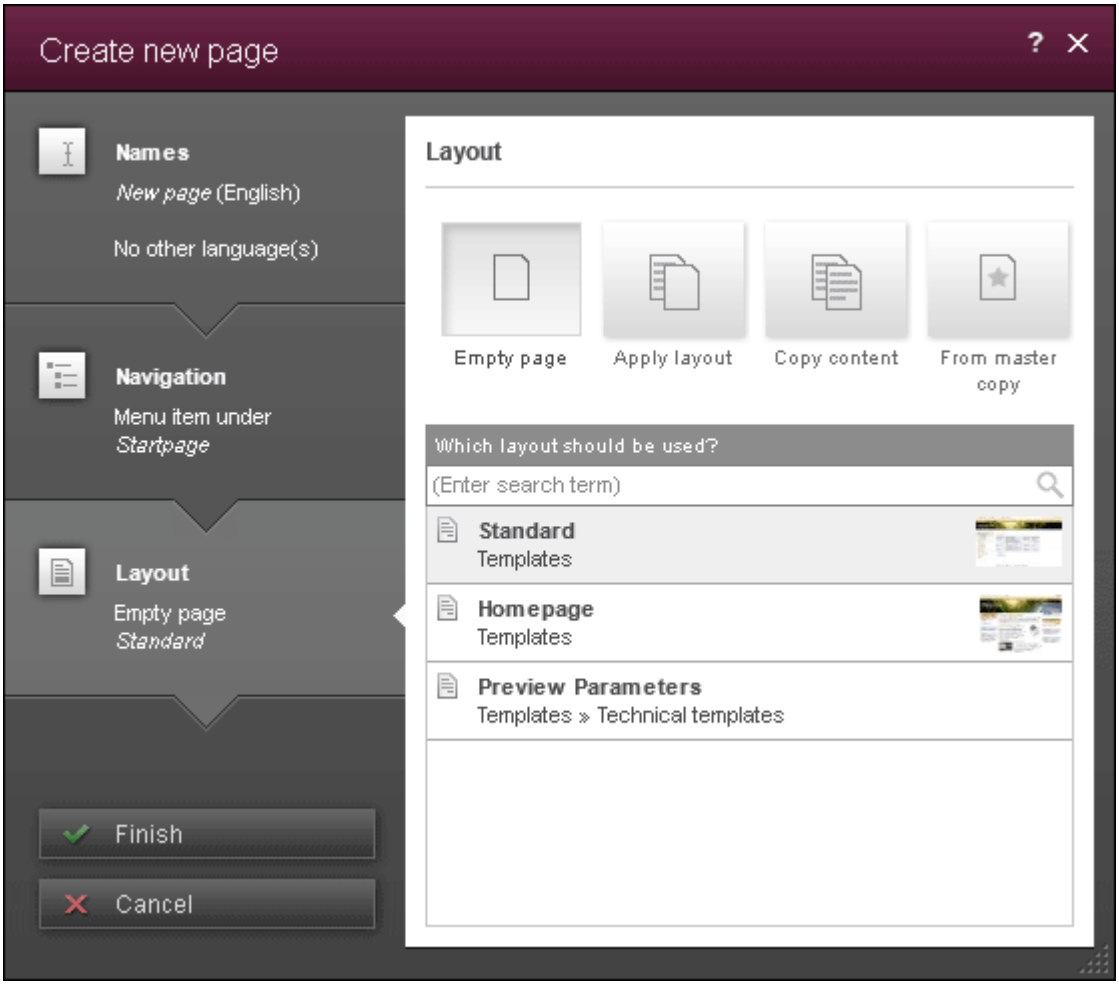


Figure 6-4: Layout selection when creating pages

- When **creating new datasets** via the "Contents" menu in the menu bar, the datasets most recently created or edited by the editor which match the relevant configuration are displayed at the top.

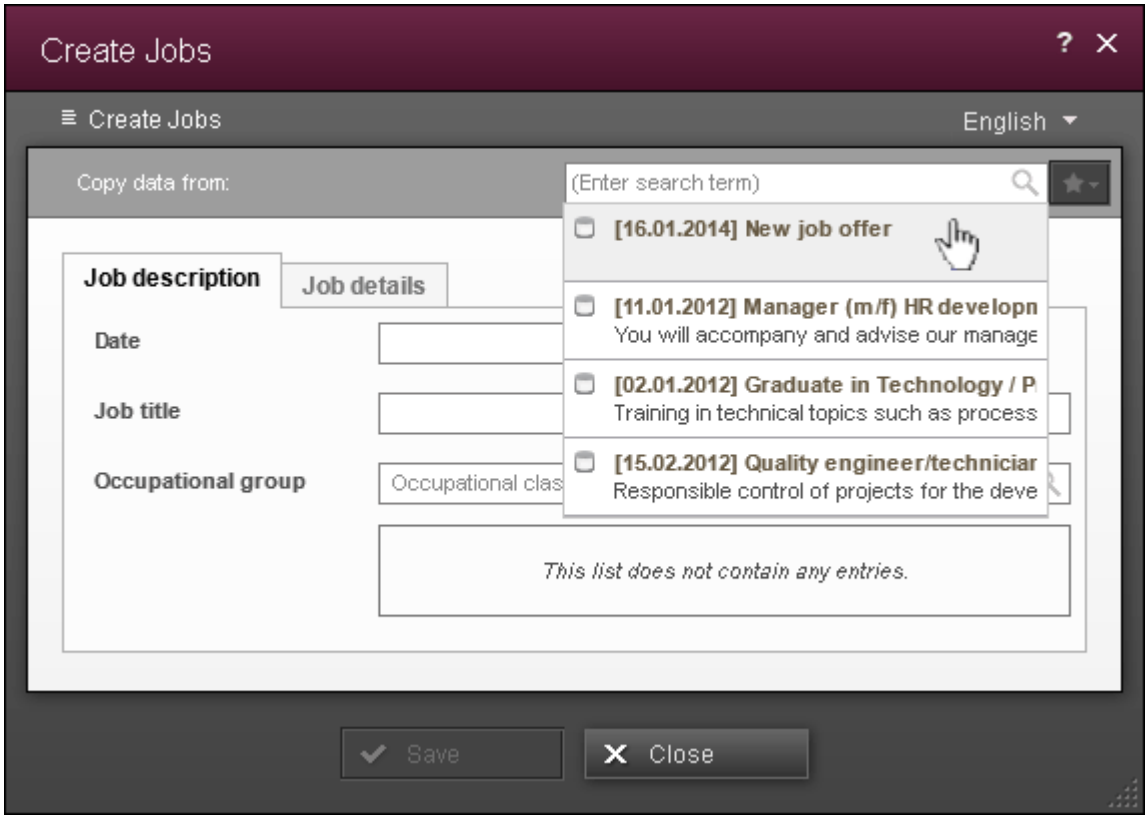


Figure 6-5: Preselection of a recently created dataset

- When **creating new sections** via the  or  icon, the templates most recently selected by the editor are displayed at the top

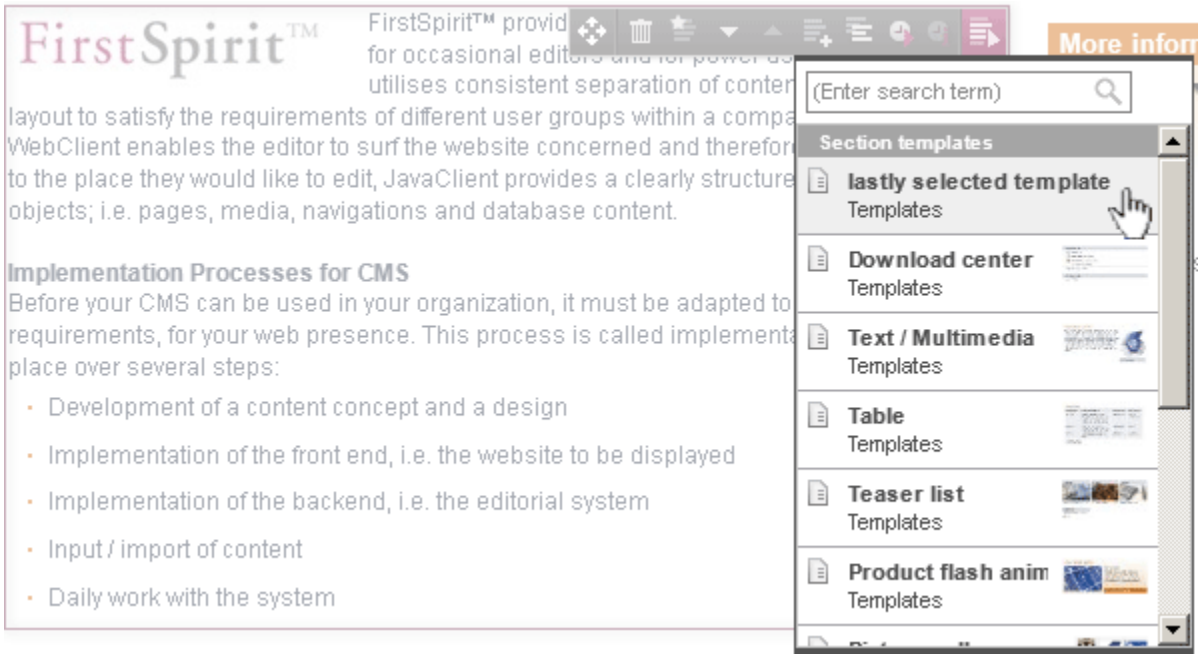


Figure 6-6: Template selection when creating a section

- When **selecting elements** (media, pages, datasets, etc.) in the integrated searches of the input elements
    - for reference selection (FS_REFERENCE)

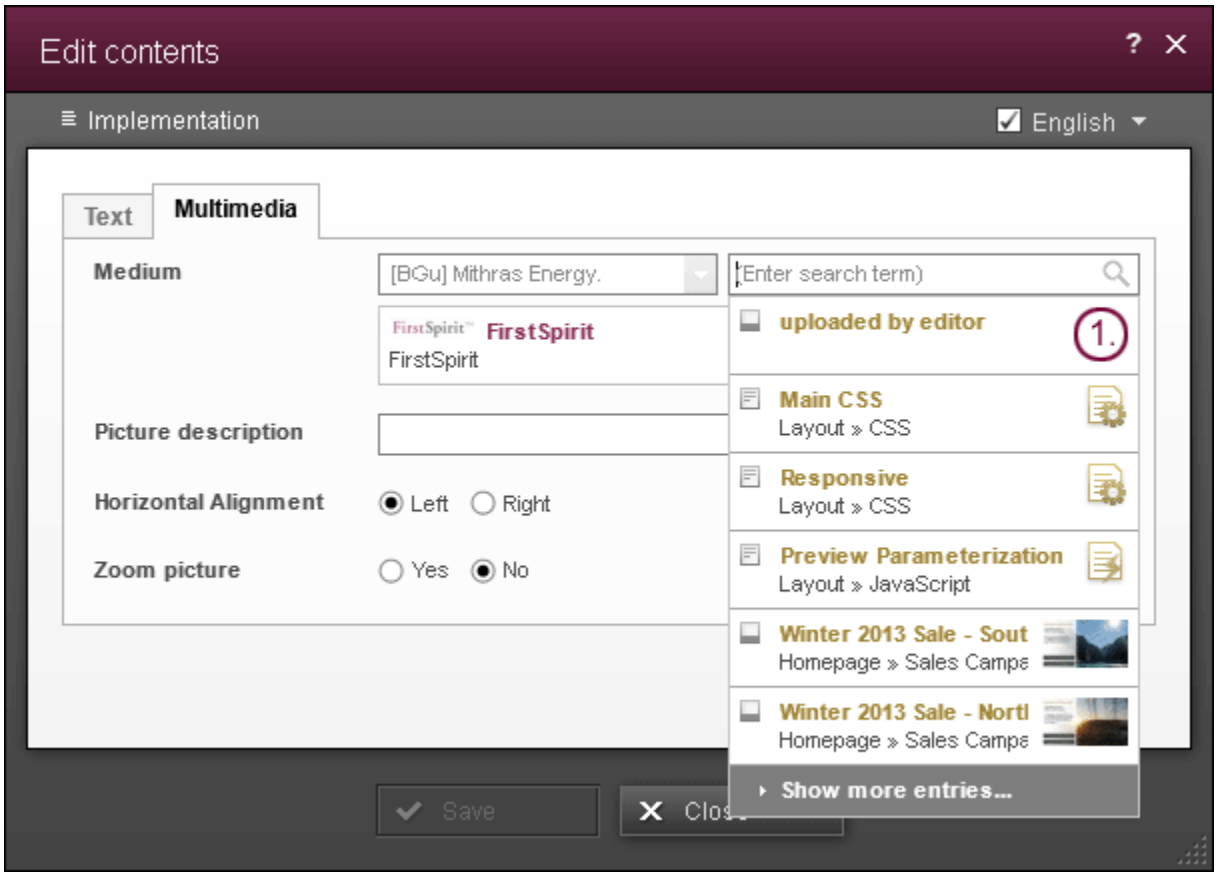


Figure 6-7: Image selection via FS_REFERENCE

- for dataset selection (FS_DATASET)

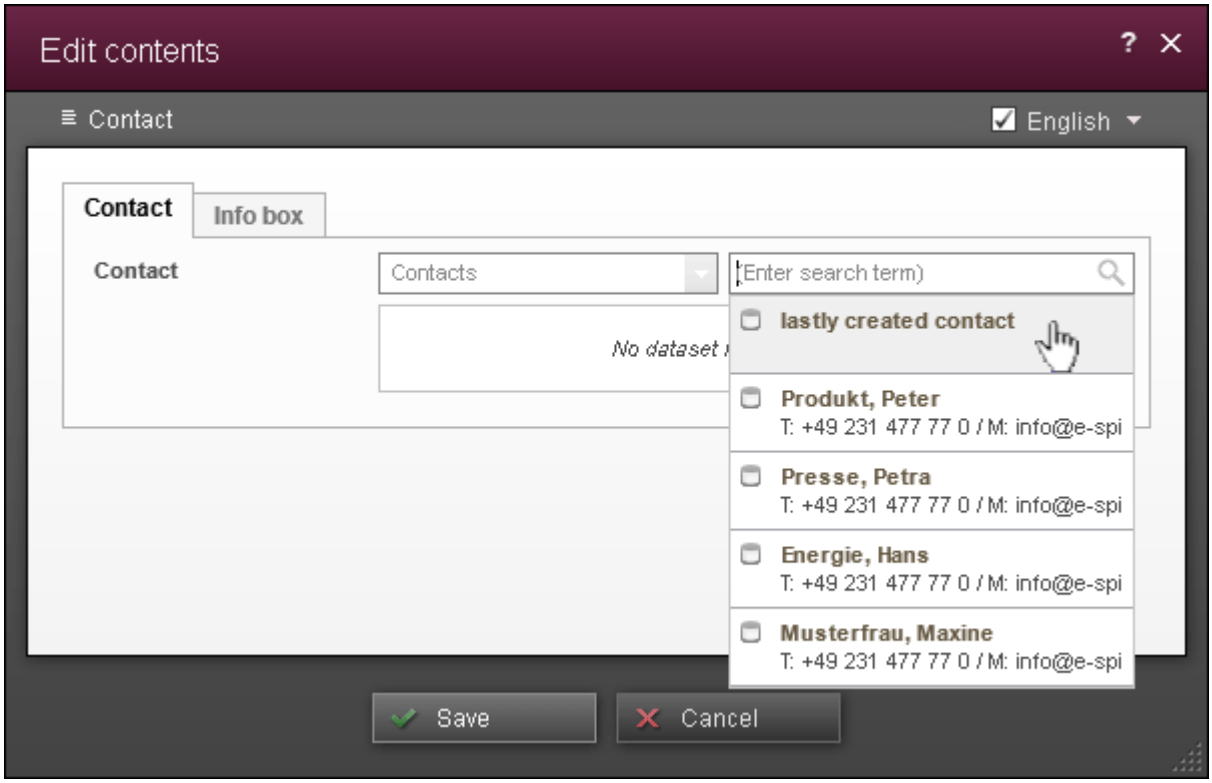


Figure 6-8: Dataset selection via FS_DATASET

the elements most recently created or edited by the editor or by other users which match the relevant configuration are displayed at the top.

The elements that have been most recently created and edited in ContentCreator also influence the "Recently used objects" displayed in SiteArchitect (e.g. when creating new pages, sections, data sources, selection dialogs, e.g. with FS_REFERENCE).

6.1.5 Optimizing work with the rich text editor

Some of the functions in the rich text editor and the rich text editor for tables have been improved/enhanced.

Whereas links could previously only be created in the rich text editor (CMS_INPUT_DOM) and the rich text editor for tables (CMS_INPUT_DOMTABLE) via the icon, they can now also be generated by using drag-and-drop to move elements to the editor, depending on the project configuration. FirstSpirit elements such as

- pages
- media
- datasets

(e.g. from search results in the Report area) can be dragged onto text that has already been entered in the editor.

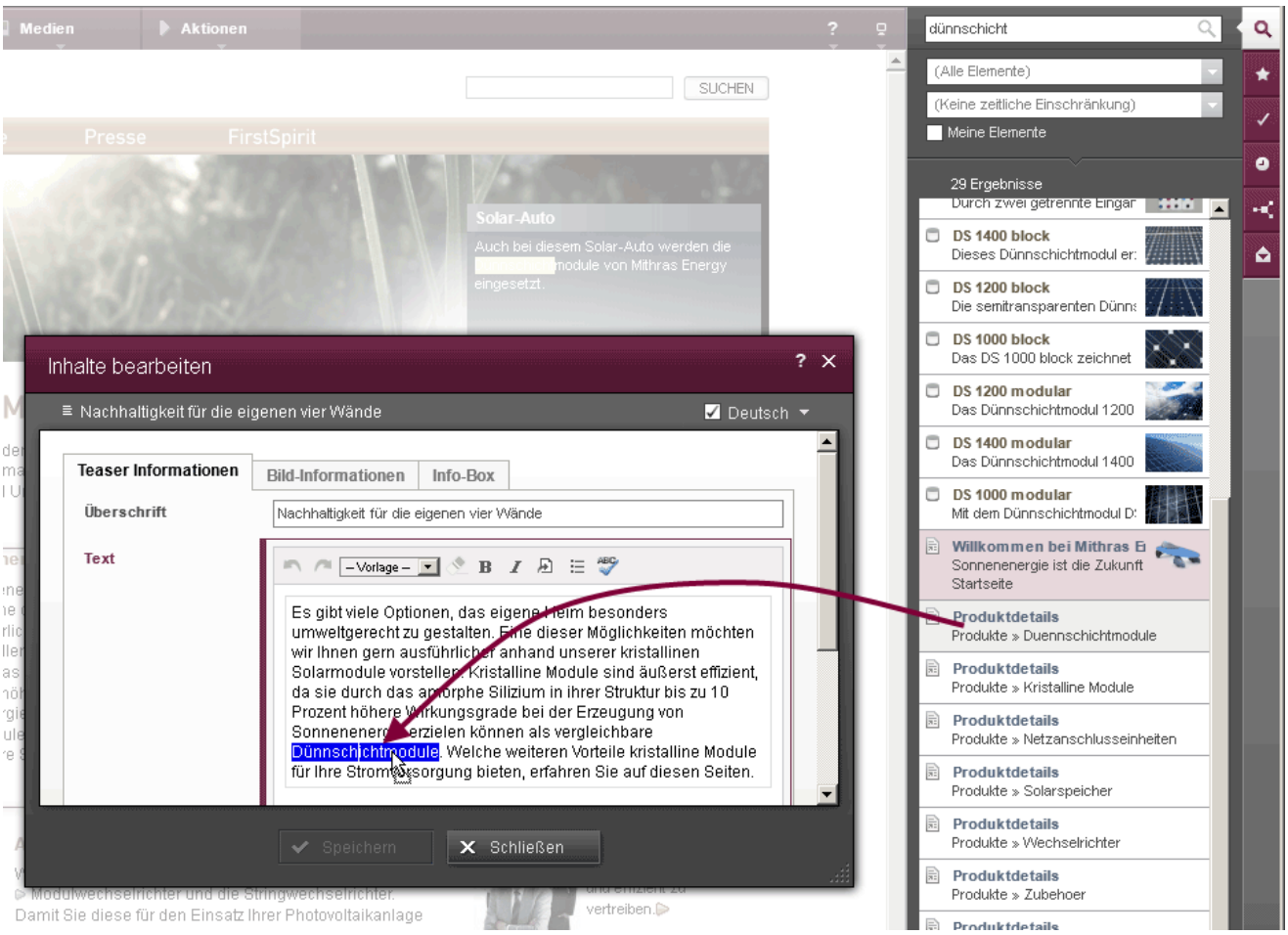


Figure 6-9: Dragging the link target (page) into the rich text editor

Color highlighting indicates the word to which the link will be assigned once the user lets go of the mouse button. In many cases, a corresponding link is created in direct relation to the type of element dropped (page, medium, or dataset). If there are several available, the types of link input on offer are displayed when the element is dropped in the editor. For example, when an image is dropped, the user can select a link input type for images ("Image link") or for pages ("Text link (internal)"):

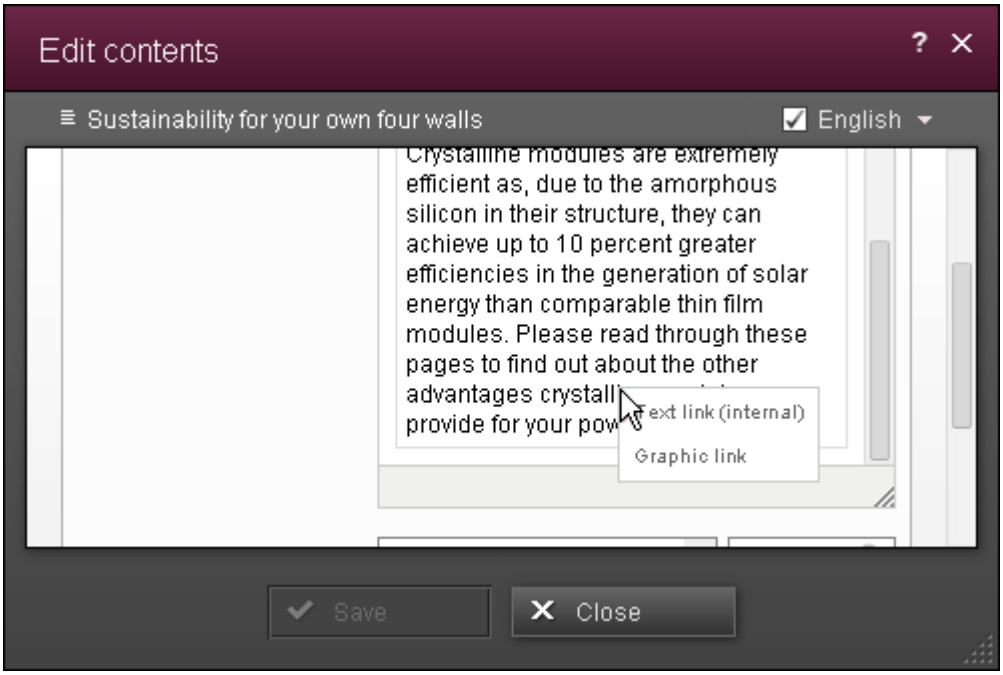


Figure 6-10: Different link creation types

With FirstSpirit version 5.1, depending on the template developer's specifications, users can click on a link to display additional information in a tooltip. Links are now highlighted more clearly.

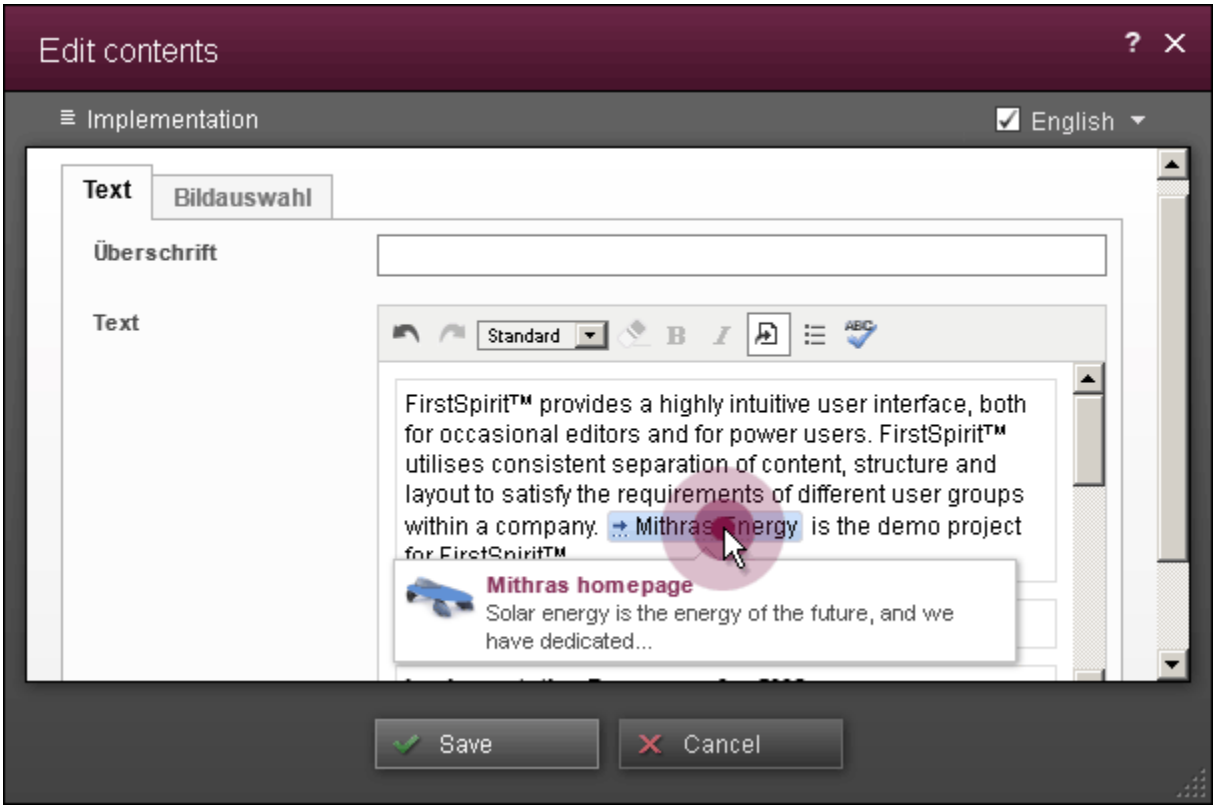


Figure 6-11: Link in rich text editor with tooltip (dialog)

Likewise when editing individual input elements:

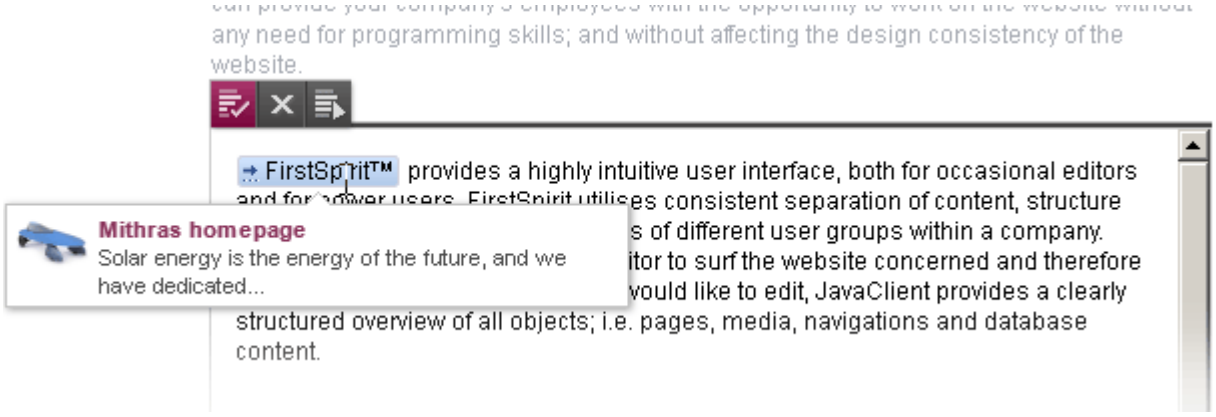


Figure 6-12: Link in rich text editor with tooltip

Links are now copied together with the information entered in the dialog that opens when the user clicks on the  icon (e.g. link text, link target, image).

For more information on links, see the online documentation on FirstSpirit ContentCreator 5, "Edit preview page"/"Input elements"/"Rich text editor", section "Add/change link", "Edit preview page"/"Input elements"/"Rich text editor for tables", section "Add/change link" and "Edit preview page"/"Input elements"/"Link input".

Handling of **lists** and **tables** has also been improved and brought into line with the procedures used in Microsoft Word. This means that <Del> can now be used to move the next section into an empty bullet point.

In version 5.1, users can navigate from cell to cell within tables using the keyboard shortcuts <Tab> and <Shift> + <Tab>. Pressing <Tab> in the last cell of the last row creates a new row below the current one.

For more information on tables, see the online documentation on FirstSpirit ContentCreator 5, "Edit preview page"/"Input elements"/"Rich text editor", section "Inline tables" and "Edit preview page"/"Input elements"/"Rich text editor for tables". For more information on lists, see the online documentation on FirstSpirit ContentCreator 5, "Edit preview page"/"Input elements"/"Rich text editor", section "Context menu for lists".

6.1.6 Checking site content from different perspectives

As Internet-enabled mobile devices such as notebooks, tablet PCs, and smartphones become more and more widespread, website designs need to be more and more flexible, with content which can be displayed perfectly on different display geometries and in different resolutions. Consequently, FirstSpirit now makes it easy for editors to check the display and navigation of website content with a variety of display sizes in the integrated preview. It also allows content, layouts, and images to be perfectly adapted to suit the output device concerned.

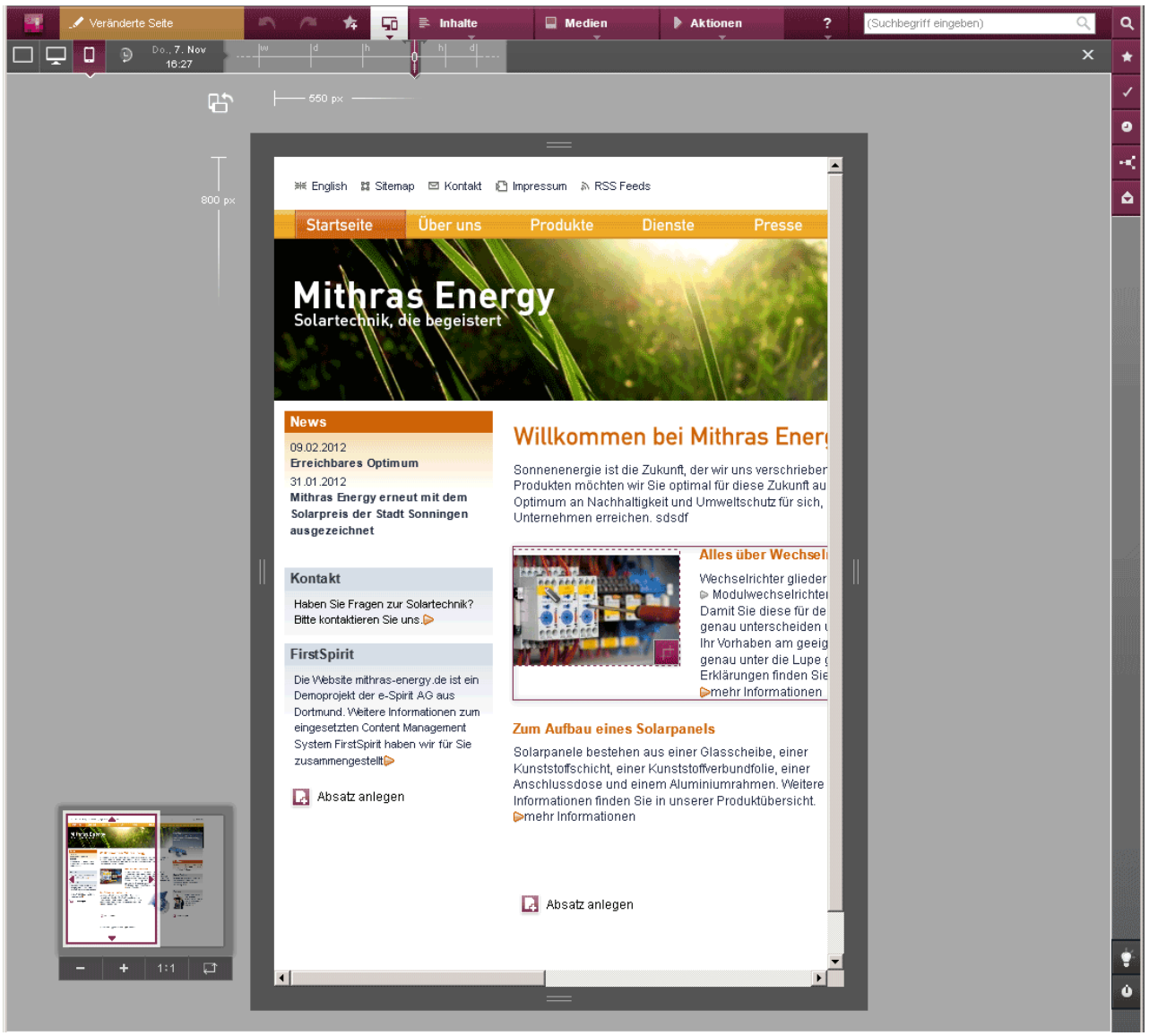


Figure 6-13: Mobile content preview

Along with size considerations, other considerations can also be taken into account, such as the page's development over time (even in the future)

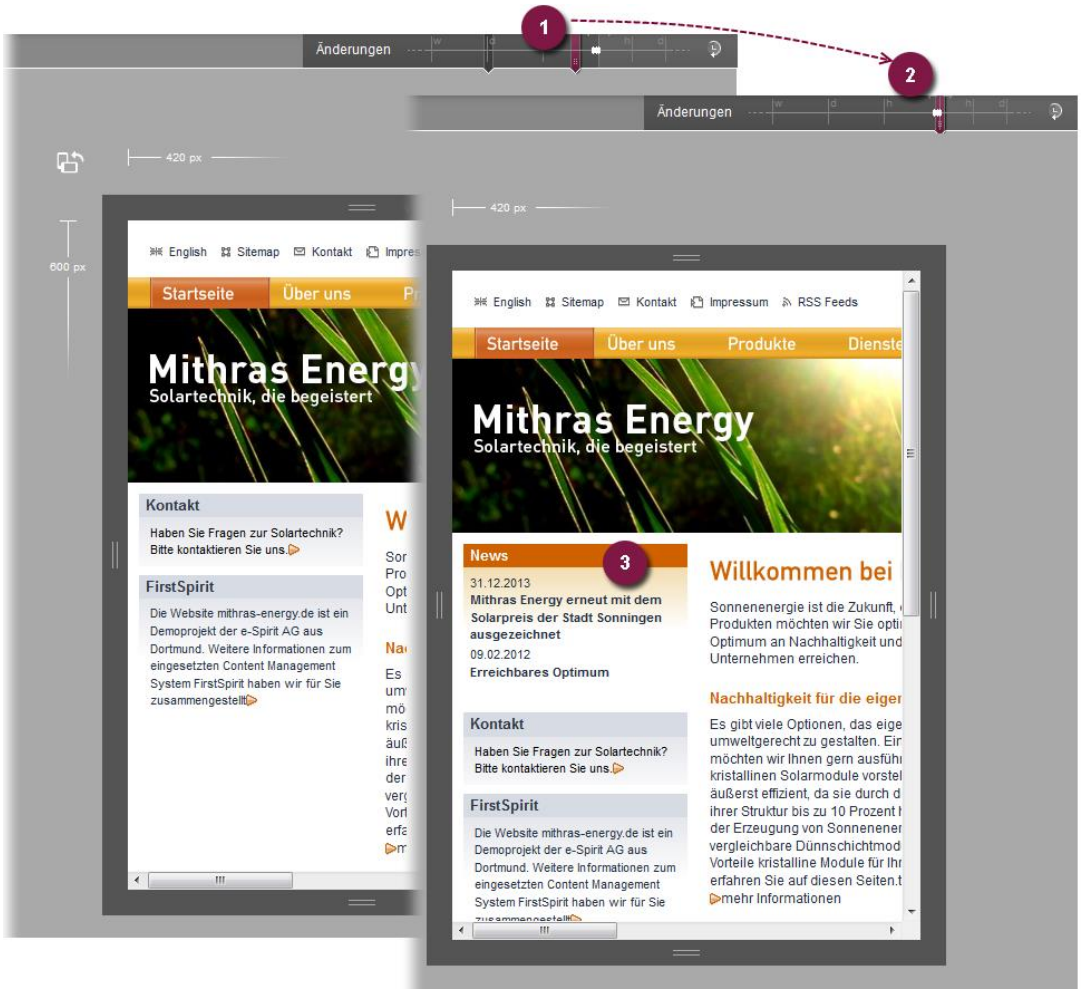


Figure 6-14: Page content at different points in time

or previews for specific user groups.

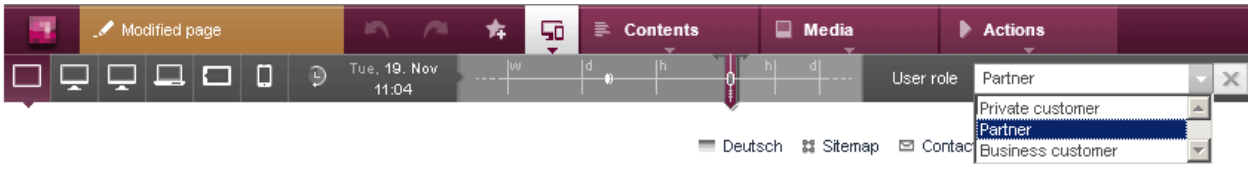


Figure 6-15: Preview for different user roles

For detailed information on setting views in ContentCreator, refer to FirstSpirit documentation on ContentCreator, "Menu functions"/"History area"/"Views". If the developer or administrator would like to configure views, refer to section 7.4.5, page 103 and section 8.3, page 118.

6.1.7 Editing content as a team and notifications

The introduction of notifications improves communication and interactions between the editors of a project. When one editor is editing the content of a site in FirstSpirit ContentCreator and the second editor attempts to edit the same content, the second editor receives a message highlighted in color (see Figure 6-16, (1)). Unlike in previous versions, an edit window appears in which the content is displayed. One editor can communicate directly with the other simply by clicking on the notification button in the message. The editor who is currently editing the content receives an automatic message in ContentCreator from the other editor, asking if they can edit the content (2).

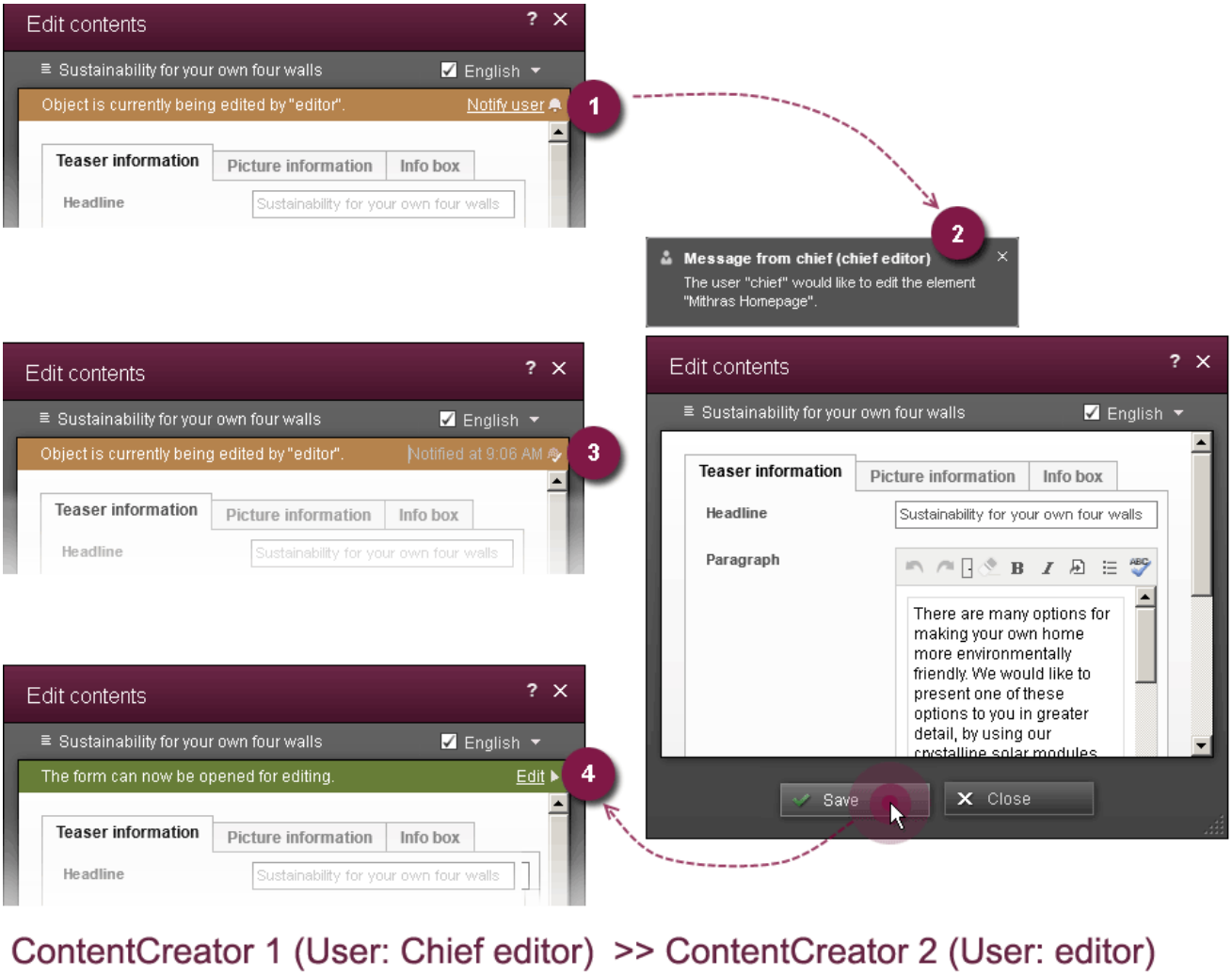


Figure 6-16: Notification function in ContentCreator

Communication is also possible in the other direction. Instead of contacting the current editor directly, the other editor can choose to be notified when the relevant content is released for editing again. In this case, ContentCreator automatically notifies the second editor that the content is available for editing again, regardless of whether the editor is on the relevant page or elsewhere in the project (see Figure 6-17).

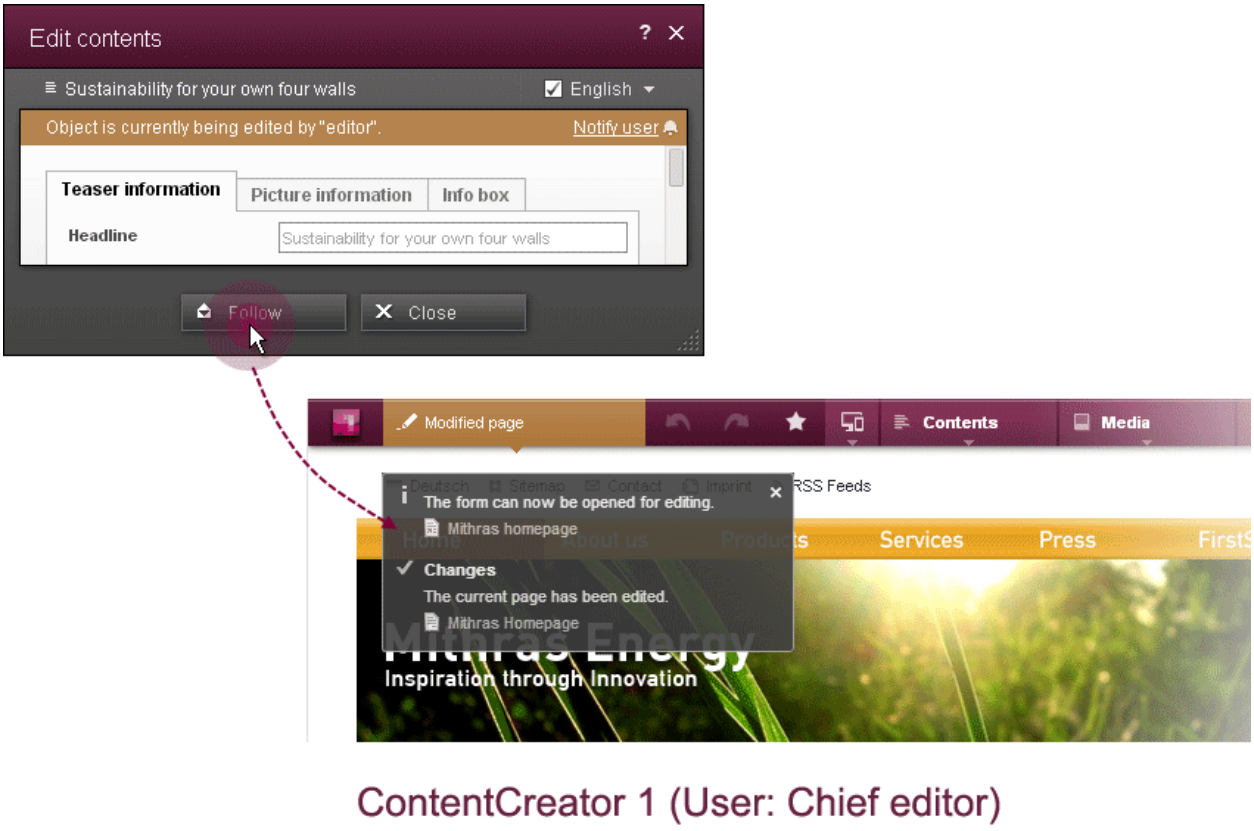


Figure 6-17: Notifications: Receiving a notification

If individual input elements are to be edited (click + <Ctrl>), the following dialog is displayed when the content is already being edited by another user:

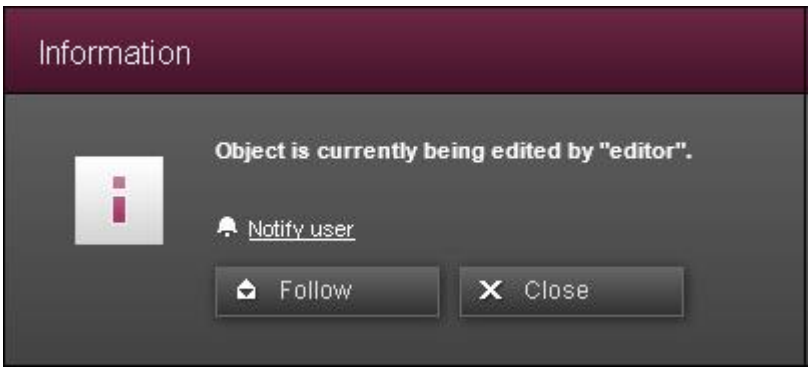


Figure 6-18: Object currently being edited

When the user clicks on "Notify user", the editor who is currently editing the object receives a message. The text changes to "Notified at 14:03" if the window is not closed. Clicking on the "Follow" button closes the edit window and editor A is notified as soon as editor B has finished editing the object.

The notification function is supported for pages and sections as well as for datasets, as long as the "Server locks in content store" option is activated for the project in the "Options" area in ServerManager.

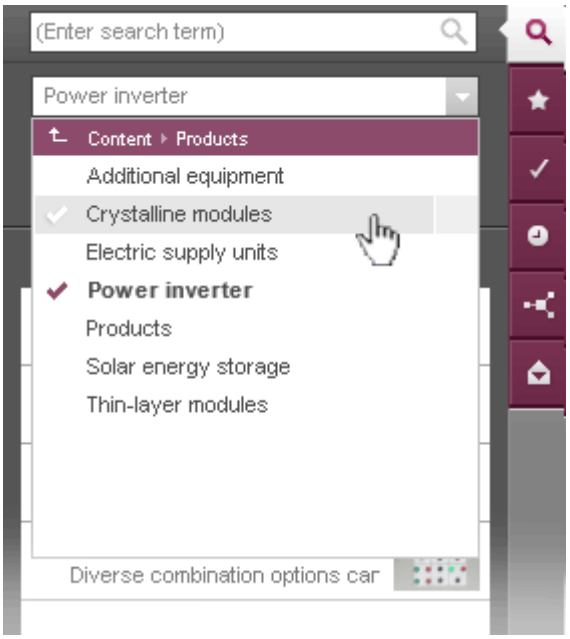
Both the direct and indirect notification functions simplify and support collaborative work within a project.

Notifications are also supported in SiteArchitect (see section 6.2.8 page 51).

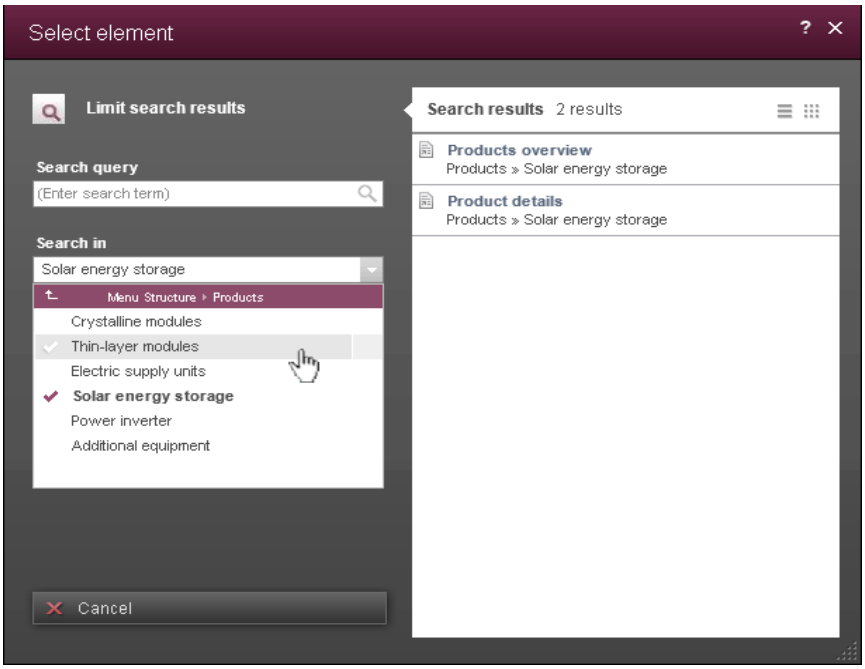
6.1.8 Further improvements

The **filter function** which filters by FirstSpirit elements in search dialogs has had a graphical overhaul, e.g.

- Search function in the Report area:



- Selection dialog in the Preview area:



To simplify **restoring deleted sections**, deleted sections are now highlighted more clearly in the project history of the Report area so that they can be restored more easily from there.

For more information on restoring deleted sections, refer also to "Report area"/"Project history" in the FirstSpirit documentation on ContentCreator.

To make the process more convenient when uploading and storing media in the project, it is now also possible to **create folders** when uploading media as of FirstSpirit version 5.1.

See also FirstSpirit documentation on ContentCreator, "Menu functions"/"Media area"/"Uploading media" and "Edit preview page"/"Input elements"/"Reference selection".

6.1.9 Tips and tricks

The FirstSpirit ContentCreator is designed so that occasional editors can use it intuitively without the need for extensive or costly training. Should questions arise, the editor can quickly obtain answers by using the introductory tour, explanatory tooltips and integrated manual included in the ContentCreator. Version 5.1 now also displays tips and tricks to editors so that they can get their work done even faster and more efficiently. This information can be disabled using the "Disable all tips" button.

6.2 New/changed functions in SiteArchitect

The SiteArchitect interface has been extensively overhauled and the design has been updated. There are also some new functions available.

New features in the Page Store, Data Store, and Media Store – the most important stores for editors – are presented below. For new features in the other stores, see section 7.3 starting from page 80.

6.2.1 Changes to FirstSpirit objects and status display

Previously, objects (except for datasets and templates) counted as changed (red text) when Edit mode had been activated and deactivated again, regardless of whether or not the content had actually been changed. In FirstSpirit version 5.1, objects only count as

changed if the content or the structure has actually been changed.

FirstSpirit elements in **Edit mode** ("Switch to Edit mode" function in the horizontal tool bar, "Edit mode on/off" function in the context menu, or <Ctrl> + E) are shown with their name in bold and with an arrow on the object icon:


In the tree structure (left-hand column) and on tabs in the workspace


In the path

If changes to the content of an element are made, a red diskette icon will appear instead of an arrow:


In the tree structure (left-hand column) and on tabs in the workspace

If changes have been made to the pages of the Page Store and/or child elements, all elements of the page that were not changed (page, sections and content areas) can be viewed using the following icon:


In the tree structure (left-hand column) and on tabs in the workspace

All elements currently in Edit mode in the current SiteArchitect can now be listed using the "Switch to Edit mode" icon in the horizontal tool bar:

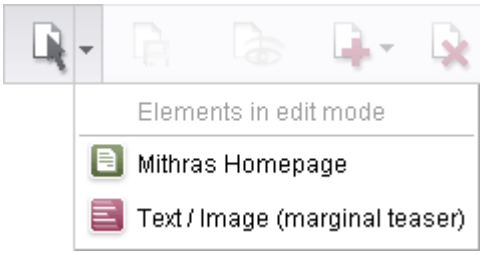


Figure 6-19: "Switch to Edit mode" icon

Users can switch to a particular element by clicking on the element. The list does not show individual sections, but rather the corresponding page.

If a workspace is closed before changes to it have been saved, the following dialog appears:

Figure 6-20: Prompt - Adopt changes?

If no change was made to an element and Edit mode was terminated for the affected workspace tab using

- the "Close workspace" context menu function on the workspace tab,
- the "Close other workspaces" context menu function, or
- the X icon on the right-hand side of the workspace tab,

the affected workspace will be closed without a prompt appearing.

Refer also to section 6.2.5.3, page 48, Figure 6-24 for more information.

Newly created elements are validated when switching to View mode (for example due to rules) and saving and quitting the Edit mode will be potentially prevented. Use "Cancel editing" (context menü "Extras" on elements of the tree structure or <Ctrl> + <Shift> + <E>) to save the element in an invalid status.

6.2.2 Project entry page

The project entry page that was previously displayed in a separate workspace tab at the start of a project is displayed on a flyout in FirstSpirit version 5.1 and can be accessed at any time via the  icon, which is now also located at the workspace tab level. The following information is displayed here at a glance, in the color of the relevant store, with the relevant object icon, object name, path, and a preview image where applicable:

- Project history: Shows the objects in the project that have been edited recently.
- Last edited: Shows the objects that have been edited recently by the current user.
- My actions: Shows the scripts for which the "Menu and entry page" scope is set (see also section 7.3.2.4 page 88).

- Task list: Shows open tasks.

Clicking on one of the displayed elements opens the relevant element directly in the workspace.

The appearance can be changed via the icon: individual columns can be hidden (by clicking on the ✓ icon) and/or the order can be changed (by clicking and dragging a column name). Deactivating the "Show at startup" entry stops the project entry page being displayed automatically when a project is next started.

If there are no elements to display, the message "(No elements found)" is displayed. Clicking on "Show report" switches to the relevant report or performs a search with the relevant search options.

The flyout remains open until it is closed again by clicking on the icon.

The license type (e.g. "Demo" or "Training") is no longer displayed on the project entry page either, but can always be found on the FirstSpirit start page.

6.2.3 Reports

In FirstSpirit version 5.1, the reports that users will know from ContentCreator can also be used in SiteArchitect. They are opened using the relevant icons in the Organize area in the left-hand client column, e.g. for an image search in the online "Fotolia" photography agency:

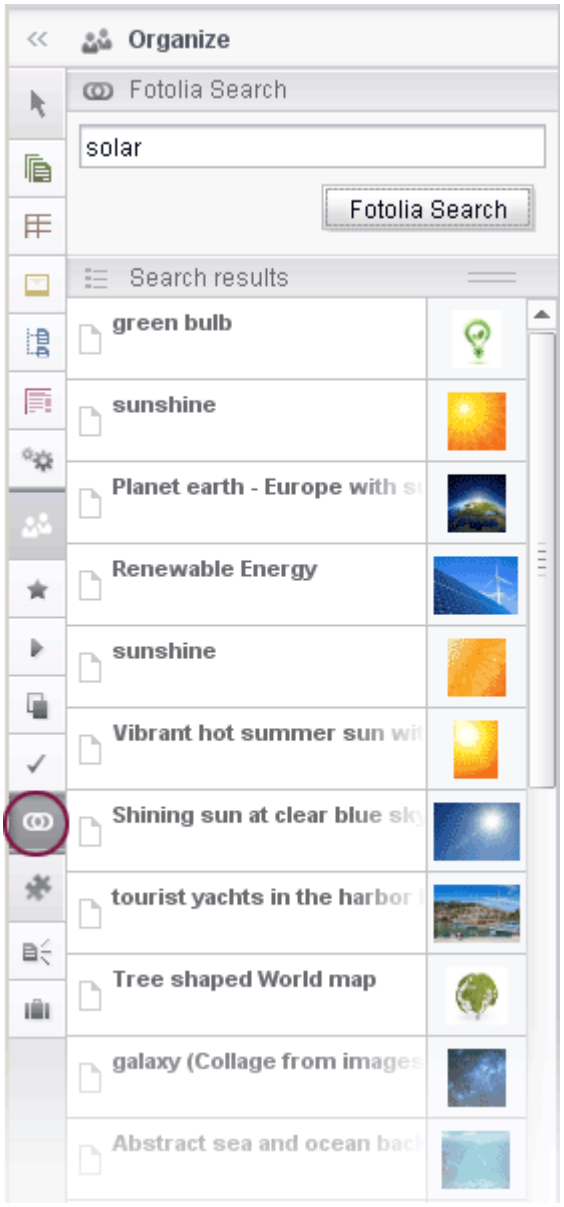


Figure 6-21: SiteArchitect with "Fotolia" report

The user enters a search term in the search field ("solar") and clicks on the "Fotolia Search" button to start the search. The user can then drag the search results onto input components in the workspace using drag-and-drop and use (reference) them there.

6.2.4 Stores: General

In order to emphasize the **color scheme** of the individual stores more strongly, the icons on the left-hand side have been modified so that the colors of the different stores

are more obvious on the icon, thus making it easier to identify the stores via the icons. The full names of the stores now appear more quickly when the user hovers the mouse over the vertical tool bar.

When FirstSpirit objects are created via the **Copy templates** function (using the "New" context menu entry or the "New" function arrow in the horizontal tool bar ), they are always in Edit mode as of FirstSpirit version 5.1 and can, therefore, be edited directly (exceptions: data sources, folders in the Data Store, folders in the Template Store).

The Media Store and Template Store offer the **Customize layout**  option which can be used to perform different functions depending on the context. See also

- Section 6.2.6, page 49 and
- Section 7.3.2, page 82.

The appearance of the **Message Board** (mostly at the root and folder levels), which users can use to communicate with each other, has also been updated.

6.2.5 Data Store

Instead of the view with the table/dataset overview at the top of the workspace and the detail view with information on the dataset at the bottom as used previously, the workspace now displays either the overview (see Figure 6-22) or the detail view (see Figure 6-23). The overview is shown initially:

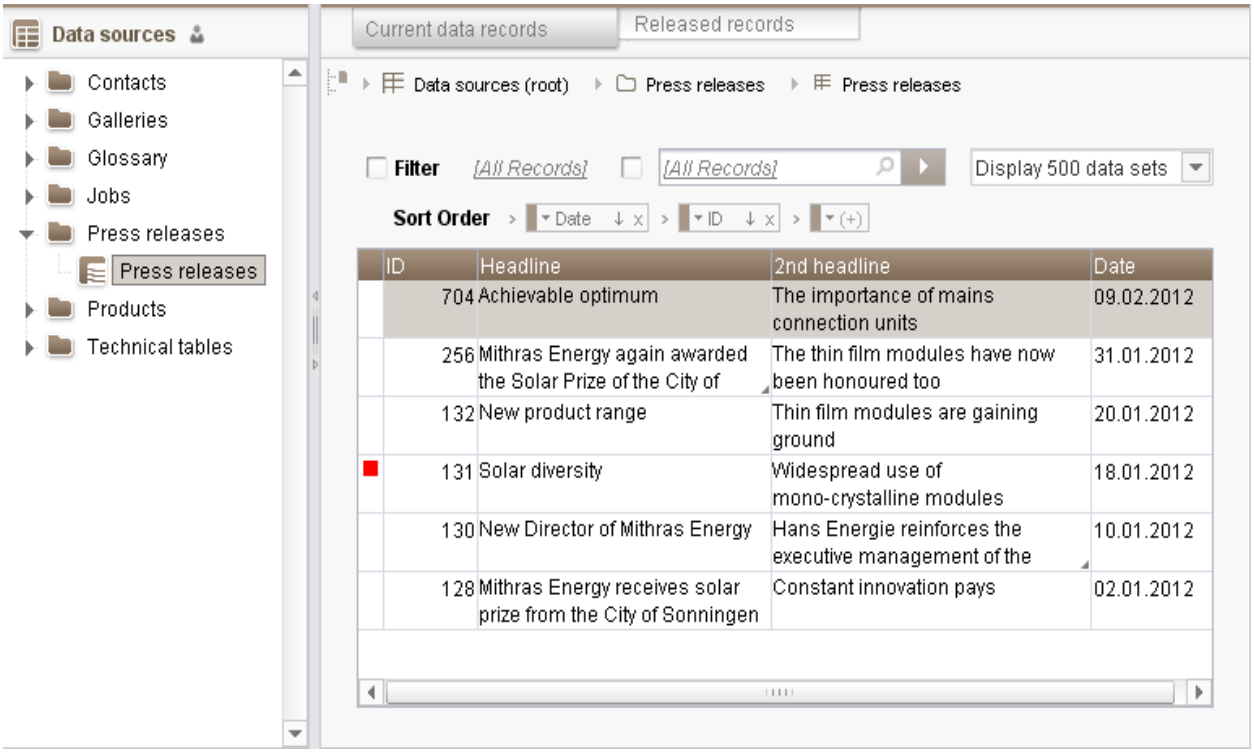


Figure 6-22: Table overview

In this view, new datasets can be created and existing ones edited or removed by using

- the relevant context menu entries on an existing dataset:
  - "New – Create dataset"
  - "New – Create a copy of dataset"
  - "Edit mode on/off"
  - "Delete"
- the relevant icons in the horizontal tool bar
  - 
  - 
  - 

  or
- keyboard shortcuts
  - <Ctrl> + N
  - <Ctrl> + E
  - <Del>

New datasets and copies of the currently selected dataset can now also be created using the context menu on the **data source node** (in the tree structure). The context menu entry "Edit mode on/off" on a data source always refers to the dataset currently selected in the overview.

Newly created datasets are in Edit mode and can be edited directly.

New functions have been added to the context menu for **datasets** in the overview, e.g.

- Display in current workspace
- Display in new workspace
- Preview / (release)
- Preview errors / (release)
- Extras
    - View template
    - Show usages
    - Display properties
    - Cancel editing
    - Display dependencies

These functions are described in the *FirstSpirit SiteArchitect documentation*, in the section "Context menus in SiteArchitect".

6.2.5.1 New detail view

The form fields of the datasets can be edited in the detail view. This view always opens when

- a dataset switches to Edit mode ( icon, context menu "Edit mode on/off", <Ctrl> + E)
- a new dataset is created ( icon, context menu "New", <Ctrl> + N) or
- the user selects a dataset in the overview by double-clicking (see Figure 6-22).

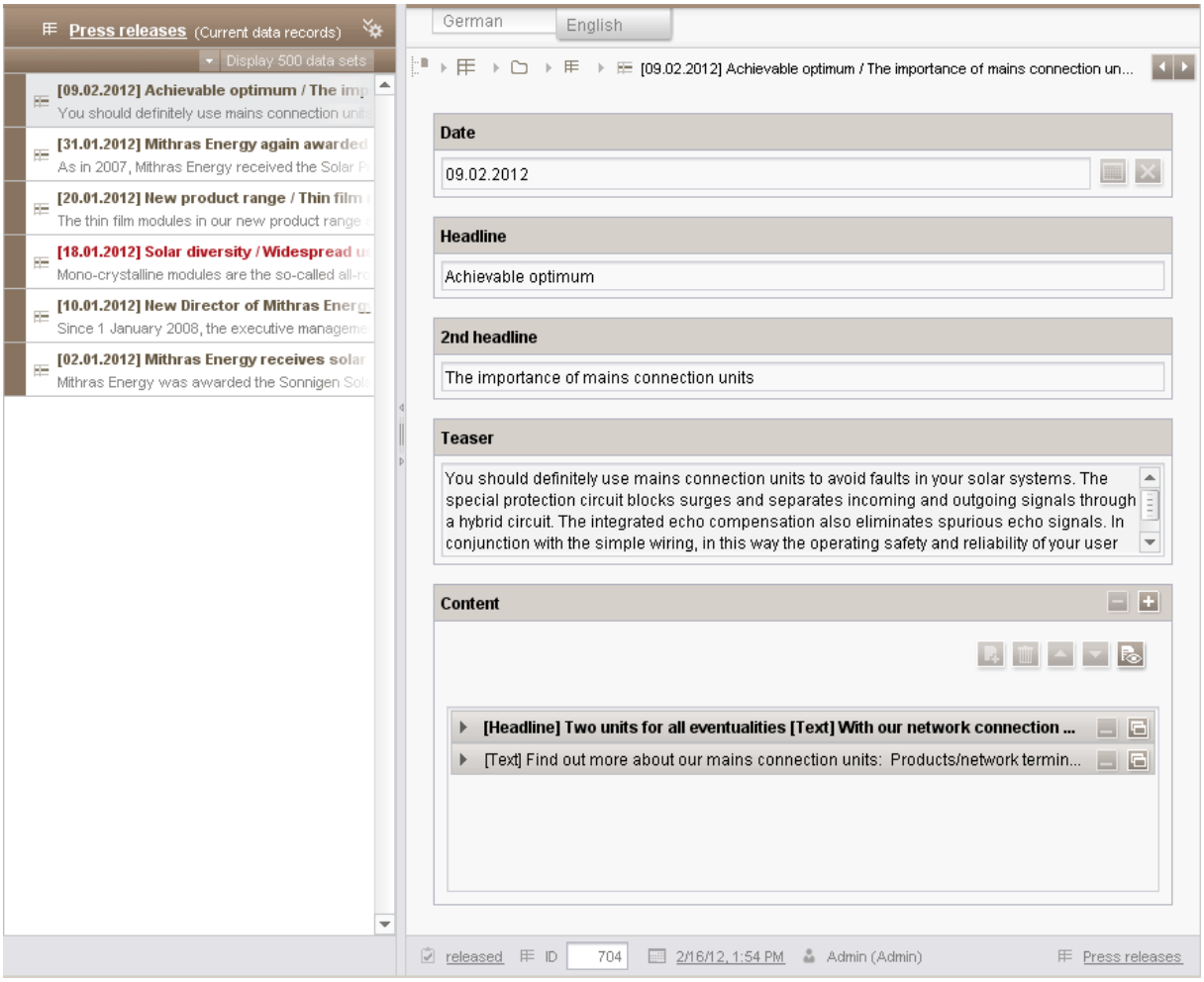


Figure 6-23: Detail view of a dataset

The dataset can be edited here using <Ctrl> + E. New datasets can be created using <Ctrl> + N in both the overview and the detail view.

Brief overview

Instead of the tree structure, a brief overview of the current datasets is now shown on the left-hand side of the screen (to configure the information shown in the brief overview, see also section 7.3.4 page 97). This brief overview has a similar structure to the main overview:

- 

The overview shows the name of the selected data source. Clicking on the name takes the user back to the main overview for this data source. The information in parentheses after the name shows whether the user is currently in the tab for the released datasets or the current datasets.

- Filter, search, and sorting options can be shown/hidden and edited using the   icons next to the heading (see section 6.2.5.2 page 47).
- The selected dataset is shown in gray in the list of displayed datasets. When one dataset is selected (by clicking, shown with a dashed frame), the user can navigate to the previous/next dataset in the list using the up/down cursor button. The desired dataset is displayed in the workspace by pressing <Return> or <Space>. The keyboard shortcuts <Ctrl> + <Page down> and <Ctrl> + <Page up> can be used during the editing process to switch straight to the next/previous dataset in the list.

Detailed information

The detailed content, including all input forms, is now displayed in the editing area for the selected dataset. Each dataset can be displayed on a separate workspace tab.  and are used as object icons here and in other places (e.g. workspace tab, new menu entries, bookmarks).

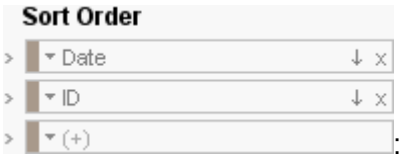
The path to the selected dataset is shown above the detailed information. The user can also switch to a different position in the tree structure via this path.

With the  "Previous/Next" icons and the keyboard shortcuts <Ctrl> + <Page up> and <Ctrl> + <Page down>, the user can easily scroll through the datasets displayed in the brief overview and edit them one after another.

6.2.5.2 Sorting, filtering, searching

In both views, the datasets can be filtered and sorted and searches can be performed:

 Filter *[All Records]*: When the user clicks on "All records", a dialog opens for filtering datasets (see "Set filter (data source)" in the *FirstSpirit SiteArchitect* documentation).



: In this area, the displayed datasets can be sorted by the various table columns (in addition to the existing option where the user clicks on the relevant column header). It is also possible to combine several columns. The datasets are sorted according to the sequence displayed for the sorting criteria. The user can change the sequence of the sorting criteria by clicking on the brown bar for the relevant sorting criterion, holding the mouse button down and dragging. The up and down arrows (↑↓) are used to set the sorting direction and the x deletes the sorting criterion. If the

user wishes to select a different column for an existing sorting criterion, this can be done by clicking on the column name. A list opens with the columns available for sorting (columns of the "FIRSTspirit Editor" type cannot be used for sorting). The user can add additional sorting criteria by clicking on the plus sign  . This also opens the list of columns available for sorting. Each column can only be selected once.

Clicking on a column header always replaces the first sorting criterion in the sequence. If the selected column was previously a sorting criterion later in the sequence, this later sorting criterion is deleted.

: Use this field to perform a full-text search. For this purpose, enter the desired search term here. Pressing <RETURN> or clicking on the  arrow icon displays the search results in the table. Additional search options are available via the  search icon next to the form field. The "Full-text search" tab has been removed from the following dialog as it is already possible to perform a full-text search using the form field that is displayed initially.

6.2.5.3 Editing datasets in series

FirstSpirit SiteArchitect is designed for editing multiple datasets one after another ("editing in series"). Changes to a dataset are saved following a confirmation prompt when the user selects the next dataset to be edited. The datasets are edited in a single workspace tab to limit the amount of mouse movement required and all important functions can be executed via a keyboard shortcut.

If a dataset in the open workspace tab is in Edit mode, the next dataset displayed in the brief overview can be selected as follows:

- By clicking on the desired dataset with the mouse
- Using the "Previous/Next"  icons
- Using the keyboard shortcuts <Ctrl> + <Page up> and <Ctrl> + <Page down>.

If **no changes** have been made to the dataset, Edit mode is deactivated for the current dataset and activated for the next one.

If **changes** have been made to the dataset, a query appears:

Figure 6-24: Query - Adopt changes?

Adopt changes: Clicking on this button saves the changes that have been made to the dataset and closes Edit mode. Edit mode is then activated for the new dataset.

Discard changes: Clicking on this button discards all changes that have been made but not saved and closes Edit mode. Edit mode is then activated for the new dataset.

Cancel: Clicking on this button keeps Edit mode open for the current dataset; Edit mode is not activated for the new dataset.

6.2.5.4 Permanently filtered data sources

Permanently filtered data sources are now shown in the tree structure with the icon. The query previously used for filtering is no longer displayed in the tree structure (see section 10.1 from page 133, entry "Displaying filtered data sources in the tree structure").

For more information on working with datasets in SiteArchitect, see also "Data Store" in the FirstSpirit SiteArchitect documentation.

6.2.6 Media Store

At the **folder level**, the functions of the former "Properties" tab can now be accessed via the icon. The following dialog opens:

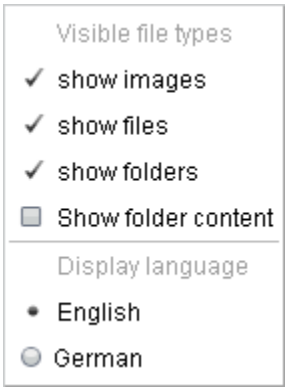


Figure 6-25: Properties at folder level

In this dialog, the user can set which contents are to be displayed on the "Overview" tab.

For detailed information on the options in this dialog, refer to the "Editing area at folder level"/"Overview tab" section of the "FirstSpirit SiteArchitect" documentation.

Media can also be sorted using the "Sort by" function. The following dialog opens:

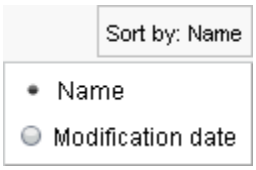


Figure 6-26: Sorting media

The sorting direction is set using the arrow.

At the **media level**, the information and functions have been rearranged slightly. For images, the icon can be used to switch between list view and the preview for the resolutions.

The **resolutions of an image** can now be sorted in the list view by the particular column by clicking on the column header, i.e. by

- resolution name ("Resolution" column, sorting alphabetically),
- image format ("File type" column, sorting alphabetically),
- dimensions ("Image size" column, sorting by the width in pixels),
- file size ("File size" column, sorting by number of kilobytes),
- status (uploaded / resized , correct resolution / incorrect resolution / resolution not yet checked).

The column width can be expanded to make it easier to read the resolution names.

6.2.7 Visualizing incorrect entries (FS_LIST)

Incorrect entries in lists (input component "FS_LIST") can now be identified more easily and quickly (as configured by the template developer), since the relevant marking is now also

- in the overview (1)
- on bars (2) and
- on tabs

displayed.

Figure 6-27: Marking incorrect entries in FS_LIST

6.2.8 Notifications

The introduction of notifications improves communication and interactions between the editors of a project. When user 1 is editing an object in FirstSpirit SiteArchitect, user 2 receives a message if they attempt to edit the same content. This message tells user 2 who is currently editing the relevant content and allows the user to notify this person.

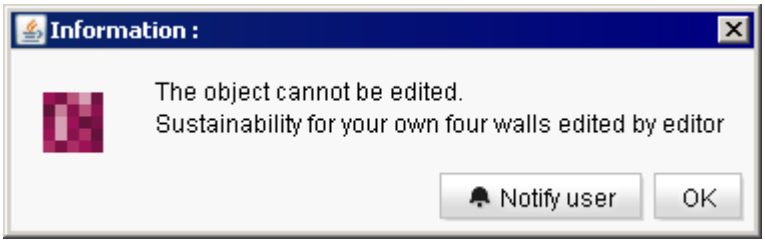


Figure 6-28: Notifying another user in SiteArchitect

To contact the other user directly, the user simply has to click the "Notify user" button. The user who is editing the current then receives an automatic message from the other user, asking if they can edit the content:

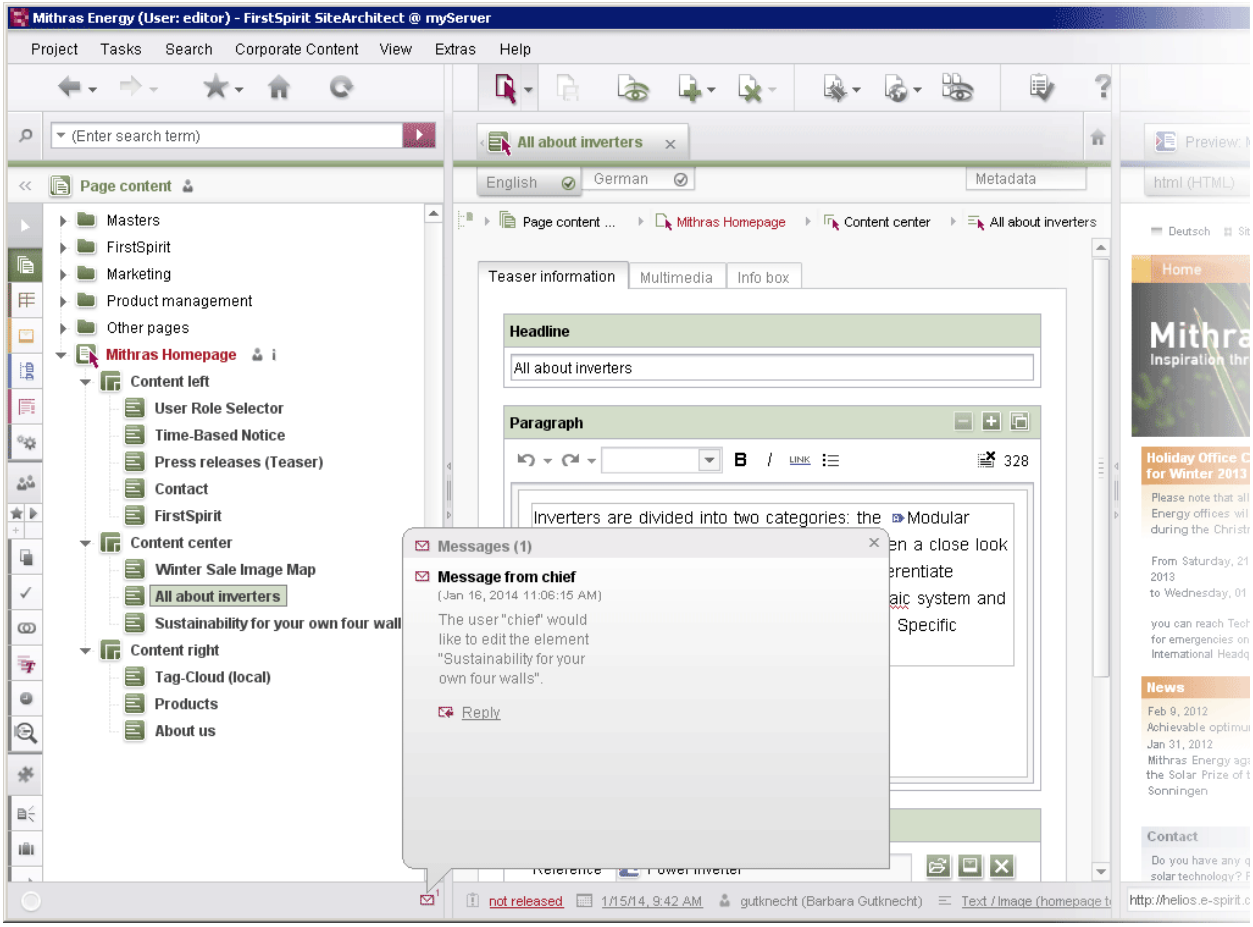


Figure 6-29: Notification in SiteArchitect

A new reply function is now available: When the user clicks "**Reply**", a "Send message" window opens. Here, the user can enter text and send the message by clicking "OK" (in the example of Figure 6-29, the message is sent to the user "chief").

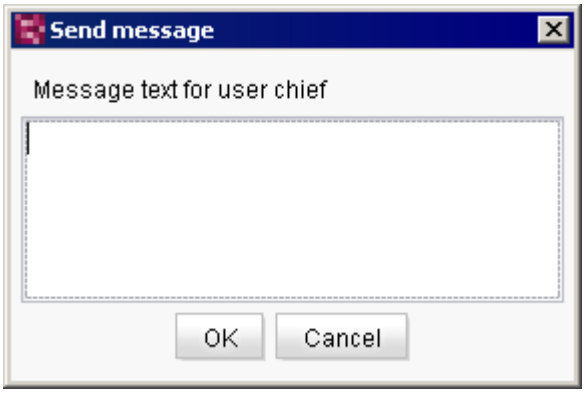


Figure 6-30: Reply function

This new reply function is also used in other places, e.g. for messages sent via

- "Extras"/"Send message" in the menu bar
- Maintenance mode

The new notifications are also supported in ContentCreator (see section 6.1.7 page 34).

6.2.9 Updates to "rich text editor" and table" input components

The list handling and formatting in the "rich text editor" (CMS_INPUT_DOM) and "table" input components have been revised in FirstSpirit 5.1. The default formats "bold" and "italic" can now for example be assigned to words without having selected the complete word. Moving list items to another level (indentation / outdentation) can now partly be achieved by using <Tab> (+ <Shift>).

For enhancements related to support for RTL script, refer also to section 5.1.4, page 17.

6.2.10 Updates to the input component for mouse-sensitive images

The input component for mouse-sensitive images (CMS_INPUT_IMAGEMAP) has been updated in FirstSpirit 5.1.

The following functions have been added in SiteArchitect:

 Jump to medium: Clicking on this icon jumps to the selected image in the Media Store.

 Remove: Clicking on this icon removes the selected image, including any frames/links that have already been created.

Furthermore, invalid references are now visualized directly in the form; for example, if the selected background image has subsequently been deleted from the Media Store or the link target has been deleted from the Site Store.

7 New functions for template developers

The "FirstSpirit Manual for Developers (Basics)" which has been available in PDF format so far was transfered into the FirstSpirit Online **Documentation** (ODFS) with FirstSpirit Version 5.1, predominantly into the sub-chapter "Templates (basics)" / "Composition of templates".

Skripting / BeanShell: Using the method `show()` you can deactivate and reactivate the output in the BeanShell console in FirstSpirit SiteArchitect. Up until now, it was necessary to activate initially the output with calling the method `show()`, it is always initially active with FirstSpirit Version 5.1. I.e., the first call of `show()` would deactivate the output in the BeanShell console.

7.1 Template Debugger and Template

The development process cycle in FirstSpirit (as well as in many other development environments) consists of "changing, testing and correcting", which only comes to a (preliminary) end after a whole series of iterations with the introduction of a new development status in the version control system. Frequently, several developers are employed to work to some extent at different locations, particularly in the case of large projects and large companies.

In order to give FirstSpirit developers the best possible support in implementing their projects quickly and cost-effectively, the following enhancements have been made in version 5.1:

7.1.1 The Template Inspector

The first steps in supporting template developers in developing HTML output in Site Architect have already been taken, including the highlighting of code text, code completion and validation. The new "Template Inspection" function has been introduced in version 5.1 to provide better orientation in existing projects and to retrieve existing code faster. Template Inspection is to the template developer what Content Highlighting is to the editor: it provides the ability to display the tag structure of the current page's HTML code in the integrated preview and from there to open the related FirstSpirit templates in the workspace:

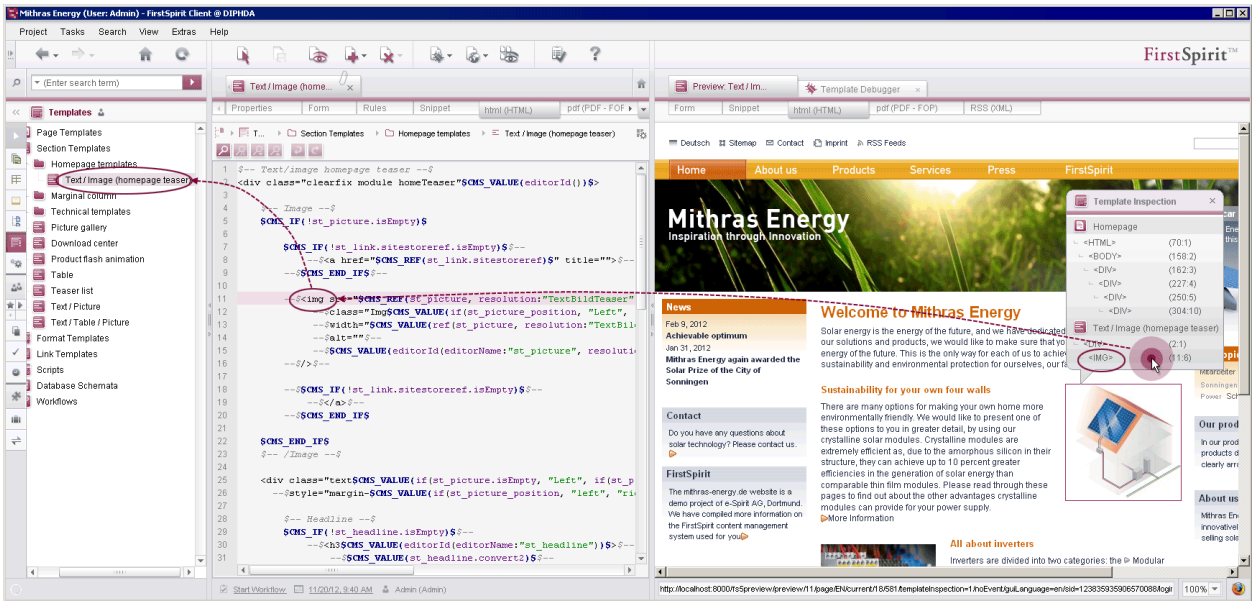


Figure 7-1: Template Inspector

The affected lines are also highlighted directly (Template Highlighting). This allows for faster retrieval and editing of locations of code in the HTML channel.

7.1.2 Bug fix and template development using the Template Debugger

The Template Debugger not only provides software-assisted troubleshooting in HTML code, but also helps the developer to develop templates and expand an existing template basis. Using the debugger, the execution paths of the template generation can be completed in detail based on the particular HTML page currently displayed in the preview. This also applies to all dependent templates (including templates for sections, tables, formats, or links).

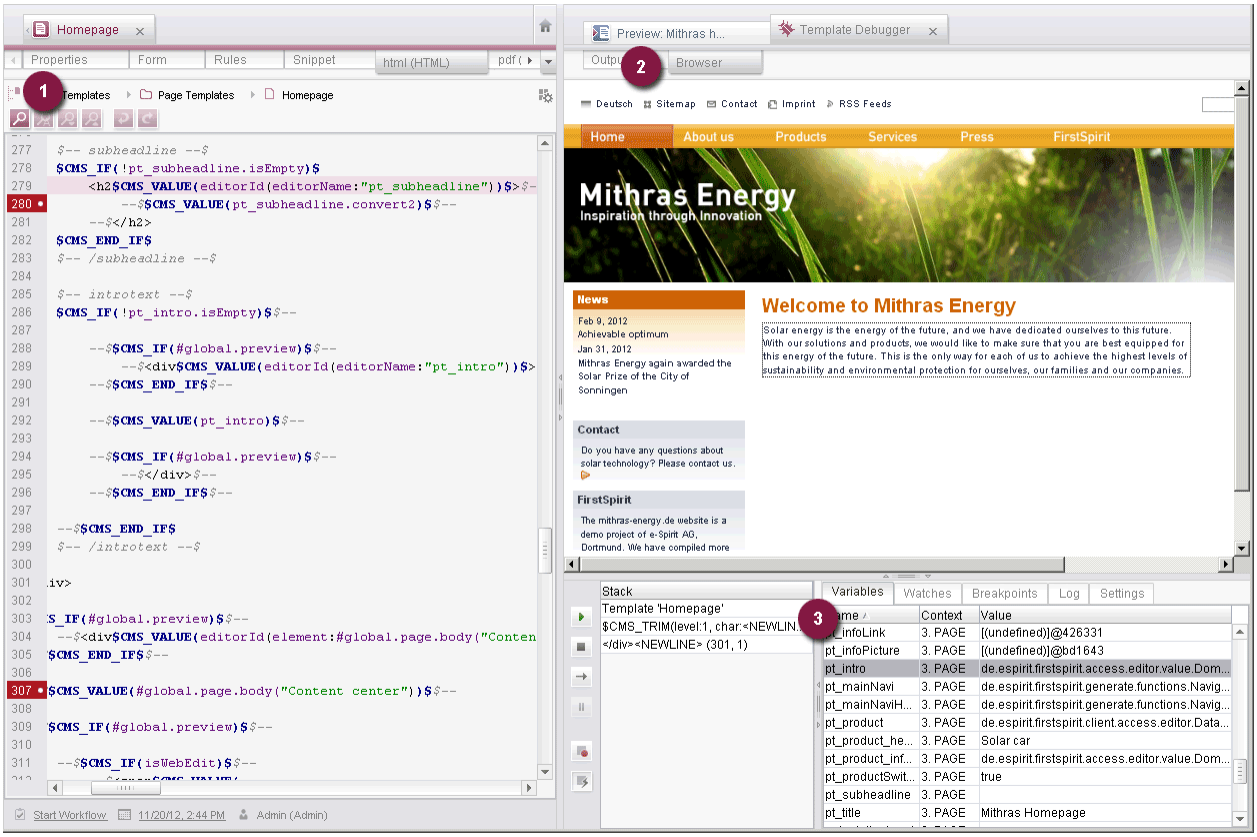


Figure 7-2: Template Debugger

In the process, a single step corresponds to an instruction in the software code. The HTML page is constructed further with each step, and the developer can then check the result either in the source text or in the inline browser ((2) in Figure 7-2). Clicking in the generated HTML then takes the developer straight to the corresponding code location in the relevant template ((1) in Figure 7-2). This procedure presents valuable insight into the inner workings of FirstSpirit syntax and the interaction between instructions, functions, variables and their output in FirstSpirit—particularly in the case of users who are new to FirstSpirit template development.

As is familiar from other debuggers, breakpoints are also defined in FirstSpirit template code. The debugger stops at these points so that the code being run through can be analyzed, thus making it possible to isolate code for troubleshooting. The integrated log output also makes troubleshooting and error analysis easier without the need to open detailed external log files first.

A key aspect of FirstSpirit template development is the use of variables. It is primarily variables that facilitate intelligent reuse as well as making content dynamic. In large projects, where many developers are working together and many variables are in use,

the number of variables can quickly become unmanageable. Which variables apply in the current context and what value do they currently have? With the Template Debugger, each developer immediately has all of the information on variables for the currently generated page or the page displayed in the preview with the current value at his disposal ((3) in Figure 7-2) and thereby has easier access to existing variables and the ability to edit them. The current value of a variable can also be checked directly on the HTML tab where the value is displayed via the variable's tooltip. In addition, the methods related to the variables are shown, including brief documentation.

The Template Debugger also works for other output media such as PDFs, etc.

For detailed information on using the Template Debugger and Inspector, see FirstSpirit online documentation, "Template development"/"Debugging"/"Where is the error?"

7.2 External synchronization of FirstSpirit files

Assembly and deployment on a FirstSpirit server is key in the development of FirstSpirit templates, modules and application integrations. To check the results, usually an instance of SiteArchitect also has to be started. This time-consuming process can be reduced significantly by providing a development environment of FirstSpirit components in which all of the necessary components of both the FirstSpirit server and SiteArchitect (including specific test functions) are brought together. The goal is that the developer can simply call up a build function in the trusted IDE directly (without a restart) and test the current results in the FirstSpirit execution environment.

In FirstSpirit version 5.1, the first steps have been taken to this effect, e.g. substructures from FirstSpirit (templates, store subtrees, etc.) can be exported to a file system structure with a folder hierarchy – in a format that is "legible to humans" and "can be interpreted by IDEs": There (that is, outside of FirstSpirit), the exported files can be edited and then resynchronized with FirstSpirit. This enables the user to synchronize changes in the file system with changes in the FirstSpirit project using a process that is largely automatic. As a result, any changes made to a template in the IDE are immediately reflected in the FirstSpirit project belonging to the developer.

The objective for FirstSpirit 5.2 is to support parallel development of templates by several, possibly geographically distributed developers in teams, aided by an external version control system.

In principle, the data format used for this special export function when FirstSpirit version 5.1 is initially released is to be retained in subsequent (minor or release) versions. However, given the highly complex structure involved, the first step is to gather empirical data regarding this new function from real projects (including from customers and partners). Based on this empirical data, later versions may see a change in the data format in which even the introduction of minor changes may be incompatible. User feedback is expressly desired in this case to make it possible to respond to partner requests.

7.2.1 Configuring external synchronization

FirstSpirit project content that is to be exported or synchronized can be selected under "External synchronization" in the "Multisite management" area of SiteArchitect (refer to the vertical tool bar).

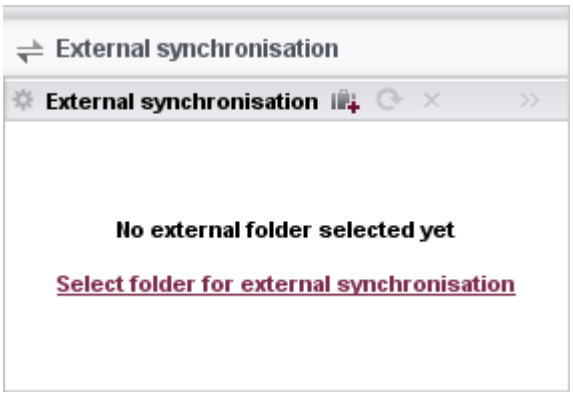


Figure 7-3: Selecting an external folder

The tool bar in this area contains entries for configuring external synchronization:

Select folder for external synchronization; this icon can be used to select an external folder in which to export the FirstSpirit objects. This could be a folder on the user's local workstation, for instance. If a folder is already selected, the icon is disabled. Alternatively, it is also possible to select the desired folder using the "Select folder for external synchronization" link. After selecting a folder for external synchronization, the basic folder path will be displayed under "FileSystem folder". The revision ID of the FirstSpirit repository is displayed under "Revision" and the date and time at which the selection was made are displayed under "Date". Imported objects and missing references are only present if the folder was already previously the target of

synchronization (including if it was the target folder for other FirstSpirit projects). After selecting the external folder, FirstSpirit project content can be added for synchronization (see section 7.2.2, page 62).

Close FileSystem folder; clicking on this icon closes the currently selected folder and with it all "Included objects" combined up to this point. A new folder can then be selected for external synchronization of FirstSpirit objects (see above).

Reload FileSystem folder; clicking on this icon loads changes to objects that have been included or exported up to this point. When changes are made in the project, the revision number and date are adapted in the "External synchronization" area, if applicable. If the setting "Synchronize automatically" is enabled in the configuration, clicking on the icon synchronizes the changed objects immediately.

» The double arrow opens a flyout menu with a detailed view of "External synchronization", showing additional information on the status of the objects to be imported or exported. A detailed description of the icons and functions used in the overview is provided in section 7.2.6, page 73.

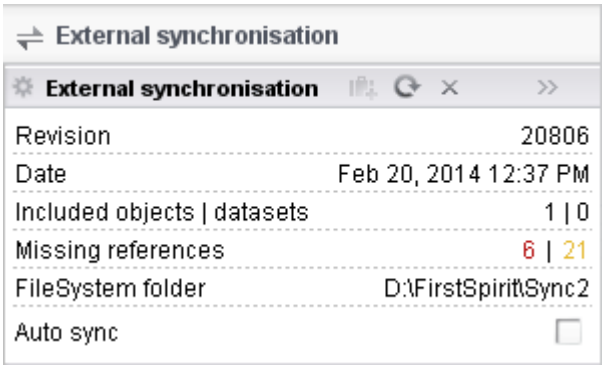


Figure 7-4: Configuration overview

Revision: Displays the revision ID. All "included objects" are exported in a revision with a number that is less than or equal to the revision number specified here. When selecting a new, external folder, the most recent FirstSpirit repository revision at this point in time is used automatically. This revision remains until it is manually updated (via the icon) or an object is included that was not yet present in the displayed revision. In both cases, the most recent FirstSpirit repository revision is updated.

Background: FirstSpirit works with a revision-based repository. A revision can be presented as a type of "Snapshot" across the entire repository at a certain point in time. In contrast to a version, which is usually only related to a single object, during a revision, the total state of all objects in the repository are listed. Revisions are listed with sequential numbering (revision ID), where there is always exactly one current revision for the whole repository. If a repository is edited, all changes carried out are linked to a new revision number. The revision number is the last current revision number of the entire repository, increased by one. All unchanged objects retain their old revision numbers. If an object is changed, it is not overwritten in the repository, but rather inserted as a new object (with a higher revision number).

Date: The date and time at which the displayed revision was added is specified here.

Included objects | datasets: This is where the user specifies the number of objects or datasets to be included in the synchronization (see section 7.2.2, page 62).

Missing references: This specifies the number of missing references for all objects that are to be exported during the next synchronization process. The number of absolutely essential objects is shown here in red; the number of optional objects is in yellow (see section 7.2.4, page 71 and section 7.2.5, page 72). Synchronization can in principle take place even if not all references have been found.

FileSystem folder: The preselected external folder is displayed here.

Auto sync: If this checkbox is activated, synchronization will start automatically if a change is made to an included object. If the checkbox is unchecked, synchronization has to be started manually each time (see section 7.2.8, page 78).

7.2.2 Adding objects

FirstSpirit objects that are to be exported or synchronized can be selected as follows:

- **Using the tree structure of the stores**
  Objects intended for external synchronization can be added directly via the tree structure of the relevant stores. There is an entry called "Add to external synchronization" for this purpose in the context menu of an object suitable for synchronization. The selected object is then *explicitly* added to the external synchronization by calling up the context menu entry. Furthermore, all of the selected object's higher level parent elements are *implicitly* added to the list of included objects. If the *explicitly* added object is a folder, all of the selected object's higher level parent elements will also be *implicitly* added to the list of included objects and all of the folder's lower level objects will be *explicitly* added.
  Even *implicitly* added objects will be synchronized in the external directory.
  The user works on a view of the data records in the Data Store when adding data records. Content sources, filtered content sources (each without datasets) or even individual data datasets can be added as objects here.

- **In the synchronization area**
  If the objects to be exported form a unit so that they can be imported into another project and function within it without issues, the required dependent objects are displayed in the "Required missing references" area (see section 7.2.4, page 71). Additional dependent objects that are not required for correct functioning of the objects to be exported are displayed in the "Optional missing references" area (see section 7.2.5, page 72). To select the desired objects for export or import, it is only necessary to select the checkbox in front of the particular object. All objects listed can be selected by activating the "Required missing references" checkbox and/or the "Optional missing references" checkbox. The objects will then be included in the synchronization process when the user clicks the "Add selected" button.

7.2.3 Included objects

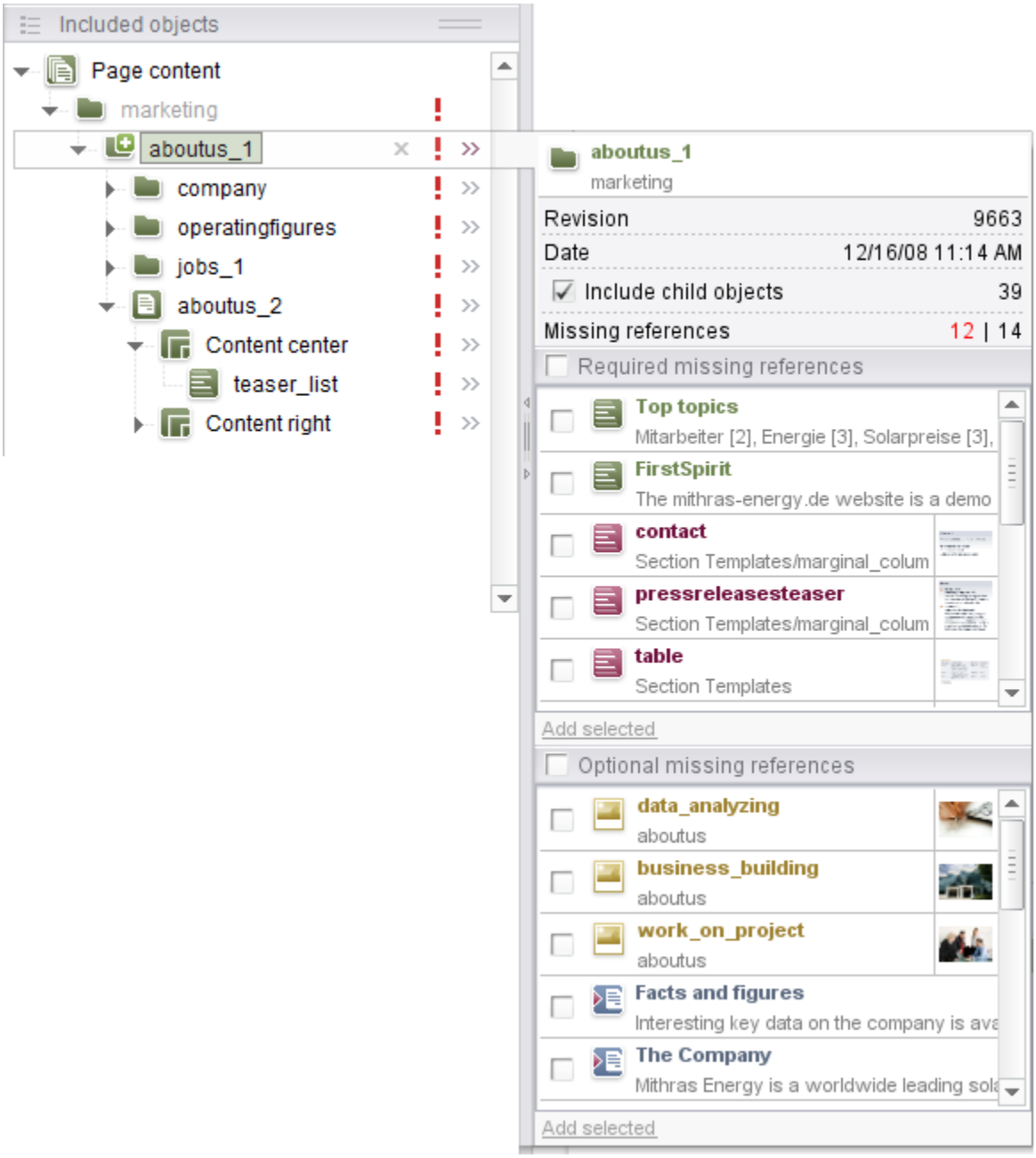


Figure 7-5: External synchronization – Included objects

All objects included for external synchronization are listed in this area. This view is always the view of the external directory and not that of the local project. This means, for example, that an element present in the external directory but not in the local project will be displayed in this view. The only exception to this are objects marked as "new".

These objects are only present in the local project and not in the external folder, but are still displayed under "Included objects".

There is a distinction between explicitly and implicitly added objects. If an object is added explicitly, then all super-ordinated objects are implicitly added to the synchronization as well (see section 7.2.2, page 62).

- Explicitly added objects are shown in normal text; this indicates that they can be removed from the list again by using the ✕ icon.
- Implicitly added objects are shown in text with less contrast and cannot be removed from the list.

Regardless, all objects in this view will be synchronized—including those that were implicitly added.

The following information is displayed for the included objects:

- Object type icon
- Synchronization state: The state is indicated by an extra icon on the object type icon (refer to section 7.2.3.1, page 65 for more information).
- Object name

In addition, hovering the mouse pointer over an included object will reveal a **Tooltip** with additional information.

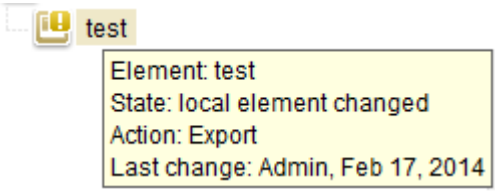


Figure 7-6: Tooltip (state and action)

In addition to the name of the object, the current state of the object, the action to be performed during the next synchronization process and the point in time when the last change was made to the object are described (refer to section 7.2.3.1, page 65 for more information).

The edit and display options are shown only for explicitly and newly added objects via the icons in the right-hand area of the "Included objects" view (all objects with the synchronization state).

- ✕ Clicking this icon removes the selected, explicitly added object from the list along with all implicitly included child objects, after the user affirms a confirmation prompt. Higher level objects that are not used by other explicitly included objects are also removed. This function removes only objects from external synchronization and not from the project or external file system. To delete objects in the file system permanently, use the "Delete external element" context menu entry on the object (see section 7.2.3.3, page 69).
- ! Required, missing references; the red exclamation mark indicates that the respective object or a child object has missing required references. The objects are listed in detail in the "Required missing references" area.
- ! Missing optional references; the yellow exclamation mark indicates that the respective object or a child object has missing optional references. The objects are listed in detail in the "Optional missing references" area.
- » Object details; clicking this icon opens a flyout menu with object-specific information (see section 7.2.3.2, page 68). Clicking the icon again closes the flyout menu.

7.2.3.1 Synchronization state and associated default actions

The state of the included object is indicated by an extra icon on the object type icon.

Synchronization state:

- State: Unchanged (already exported object without local or external changes).
- State: New (object newly added locally and not previously exported).
- State: Element not found (object newly added externally and not previously imported).
- State: Local element changed (object already exported has been changed locally).
- State: Local element changed, path changed (object already exported has been moved locally).
- State: Files changed (object already exported has been changed externally).
- State: External element changed/path changed (object already exported has been moved externally).
- State: External element changed/files changed (object already exported with external change from another FirstSpirit project).

- State: Local element changed/files changed (object already exported with change made externally and locally – conflict; refer to section 7.2.7, page 75).
- State: Element not found, element deleted (object already exported was deleted locally).

A default action is linked with each synchronization state, which is carried out automatically during the next synchronization process as long as the editor does not change this preselection manually. The state and the action to be performed are displayed on the relevant object (see Figure 7-6) or via the flyout menu (see section 7.2.6, page 73).

Actions:

- Action: None
- Action: Export (object from local FirstSpirit project is exported to file system).
- Action: Import/Update (object from file system is imported to local FirstSpirit project).
- Action: Delete (exported object is deleted from file system).

Possible state/action combinations:

- State: Unchanged; Action: None
  If no external or internal changes have been made to an object already exported, the object remains in the list of included objects, but is not synchronized.
- State: New; Action: Export
  An object added for synchronization that has not yet been synchronized. This object will be exported to the external folder during the next synchronization process.
- State: Element not found; Action: Import/Update
  A new object from a different source (e.g. from a different FirstSpirit project) has been exported to the external folder. This external file will be reimported to the FirstSpirit project during the next synchronization process.
- State: Local element changed; Action: Export
  The included object has been changed locally in the FirstSpirit project since the last synchronization process. This change will be exported to the external folder during the next synchronization process.
- State: Local element changed, path changed; Action: Export
  The included object has been moved locally in the FirstSpirit project since the last synchronization process. These changes will be exported to the external folder during the next synchronization process.

- State: Files changed; Action: Import/Update
  The exported file has changed in the external folder. During the next synchronization process, the object from the external folder will be imported into the FirstSpirit project and the changed state will be applied in the project.

- State: External element changed/path changed; Action: Import/Update
  An object from another source (e.g. from a different FirstSpirit project) has been moved since synchronization. During the next synchronization process, the changes from the external folder will be imported into the FirstSpirit project and the changed state will be applied in the project.

  State: External element changed/files changed; Action: Import/Update
  An object was changed in a different source (e.g. in a different FirstSpirit project) and was exported to the external folder. The external change will be imported to the local FirstSpirit project during the next synchronization process and the changed state will be applied in the project.

- State: Local element changed/files changed; Action: None.
  Here a conflict is present, since both the exported object and the internal object have been changed in the project since the last synchronization. In this case, the user must decide which change to apply during the next synchronization process. A default action therefore does not take place (refer to Resolving conflicts in section 7.2.7, page 75).

- State: Element not found, element deleted – Action: None
  The included and already exported object was deleted from the FirstSpirit project since the last synchronization took place. In this case, the user must decide which change to apply during the next synchronization process. A default action therefore does not take place (refer to section 7.2.7, page 75).

7.2.3.2 Flyout menu: Detailed view - Included objects

The flyout menu can be opened using the » icon located after each new, explicitly added object in the "Included objects" area.

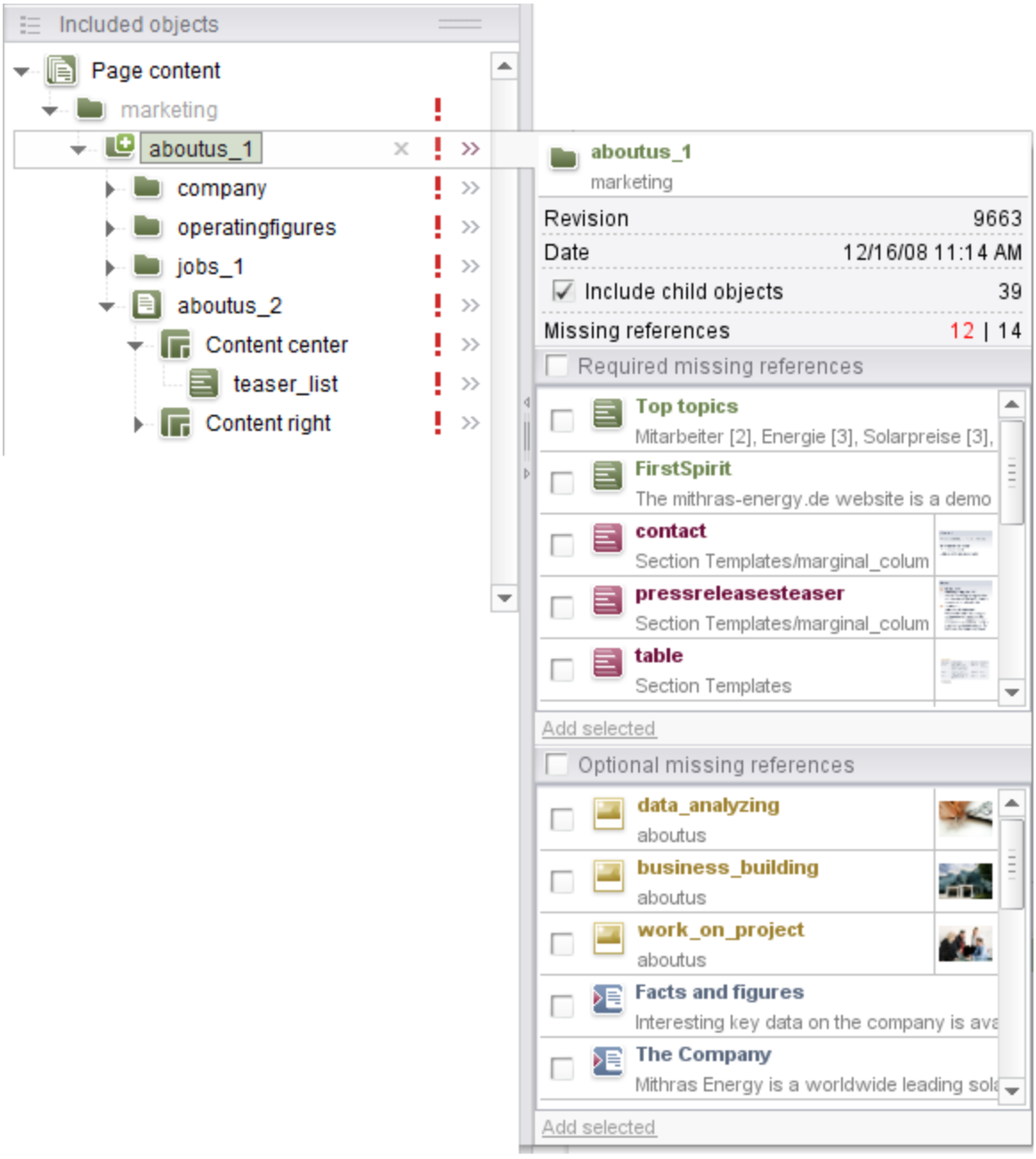


Figure 7-7: Detailed information

The flyout menu contains detailed information about the particular object:

- Icon, language-dependent display name, path or teaser of displayed object
- Revision of the object (object-specific state to take into account for the synchronization) as well as the date and time of the revision.
- "Include child objects" checkbox – If this checkbox is selected, then the missing references for the displayed object are displayed along with all of the object's child objects. If the checkbox is not selected, then only the missing references of the displayed object are shown.
- Number of child objects
- Number of missing references (required and optional)

The "Required missing references" and "Optional missing references" areas are displayed similarly to the areas with the same name in the overview (see section 7.2.4, page 71 and section 7.2.5, page 72), but here only the references (dependencies) for an object are displayed.

7.2.3.3 Context menu – Select action

An included object's context menu can be used to select the synchronization action that should take place the next time this object is synchronized. The action is preselected depending on the synchronization state of an object (see section 7.2.3.1, page 65), but can be changed using the context menu or the detailed view (see section 7.2.6, page 73). The context menu is available for all included objects that have already been synchronized externally (not for objects in the local project that have been newly added for external synchronization).

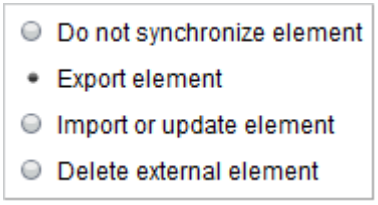


Figure 7-8: Context menu

"Do not synchronize element" action: This is the preselected default action in the case of a conflict. To prevent unintentional overwriting and deleting of objects, synchronization is prevented in the case of a conflict. The user must now decide which changes to apply in the case of a conflict (see section 7.2.7, page 75).

"Export element" action: The next time synchronization is performed, the changed version will be exported from the project to the external directory. The changes in the external directory will be overwritten.

"Import or update element" action: The next time synchronization is performed, the changed version will be imported from the external directory into the project. The internal changes in the project will be overwritten. If new objects have been added in the external directory, these objects will be imported to the local project and recreated there.

"Delete external element": The next time synchronization is performed, the object will be deleted from the external directory and will thus also be removed from the list of included objects.

Deleting external element: As long as no additional security function has been planned in the file system (e.g. connection to a version control management system), this action cannot be revoked. There is no way to undo this change using the "External synchronization" function or to restore deleted objects from the Recycle Bin.

7.2.4 Required missing references

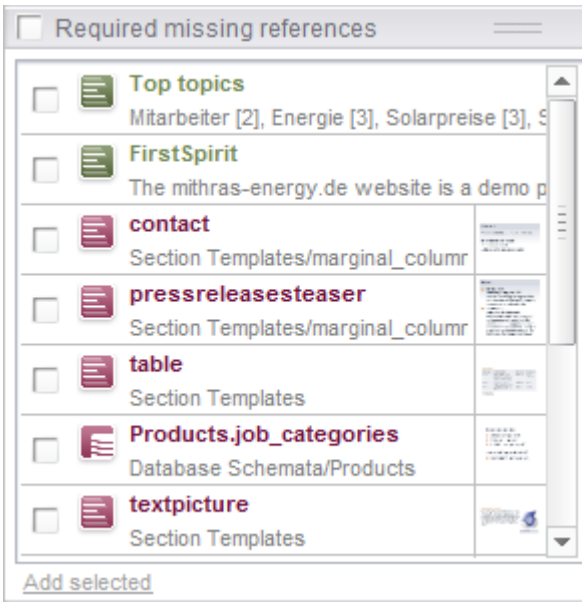


Figure 7-9: Required missing references

In this area, all objects are displayed that are required for exporting a self-contained, functional sub-area of a project. For example, if a page is exported, the templates on which this page is based are required in order to view the page when it is imported to another FirstSpirit project. The user must ensure that the necessary dependence are present in the target project.

External synchronization can then also be performed if the required dependencies have not been added for external synchronization.
If these missing references are not already found in the target project, such as during synchronization that is performed exclusively between a project and an external directory, the synchronized objects in the target project are present, but may possibly be unusable and could cause errors in the project.

Required objects are displayed in list form, each with a checkbox for selecting each individual object. Clicking on an object in the list opens a tab with that object's forms in the SiteArchitect edit area for viewing. The object cannot be edited here, since it is an historical view. This can be identified by a clock symbol on the object icon (refer to "Revision" in section 7.2.1 for more information).

If all of the required references are made, then this area remains empty.

Required missing references: If this checkbox on the top end of the area is selected, then the checkbox for selecting an object is selected for all of the objects in the list.

Clicking on the "Add selected" option includes all of the objects selected in this area for synchronization.

7.2.5 Optional missing references

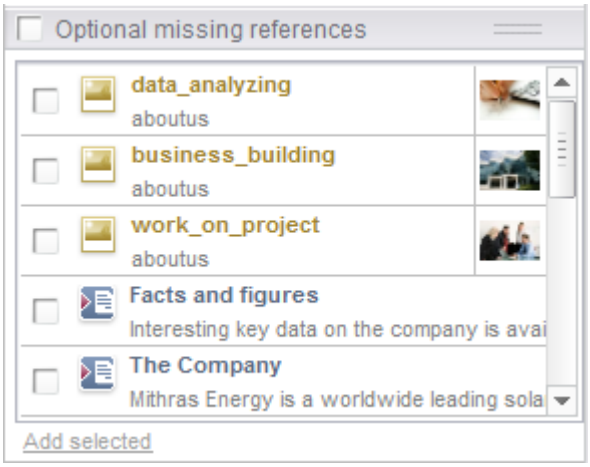


Figure 7-10: Optional missing references

In this area, all objects are displayed that are still referenced by the object already selected for export, but they are not required for the export of a self-contained, functional sub-area of a project. For example, if a page is exported, the media referenced on this page can be added as an option.

Optional objects are displayed in list form with a checkbox for selecting each individual object. Clicking on an object in the list opens a tab with that object's forms in the SiteArchitect edit area for viewing. The object cannot be edited here, since it is a historical view. This can be identified by a clock symbol on the object icon (refer to "Revision" in section 7.2.1 for more information).

If all of the optional references are made then this area remains empty.

Optional missing references: If this checkbox on the top end of the area is selected, then the checkbox for selecting an object is selected for all of the objects in the list.

Clicking on the "Add selected" button includes all of the objects selected in this area for synchronization.

7.2.6 Flyout menu: Detailed view - External synchronization

In addition to the "Included objects" view, the state of the objects to be imported or exported can also be displayed and changed using the "External synchronization" detail view. The flyout menu can be opened using the » icon in the top tool bar:

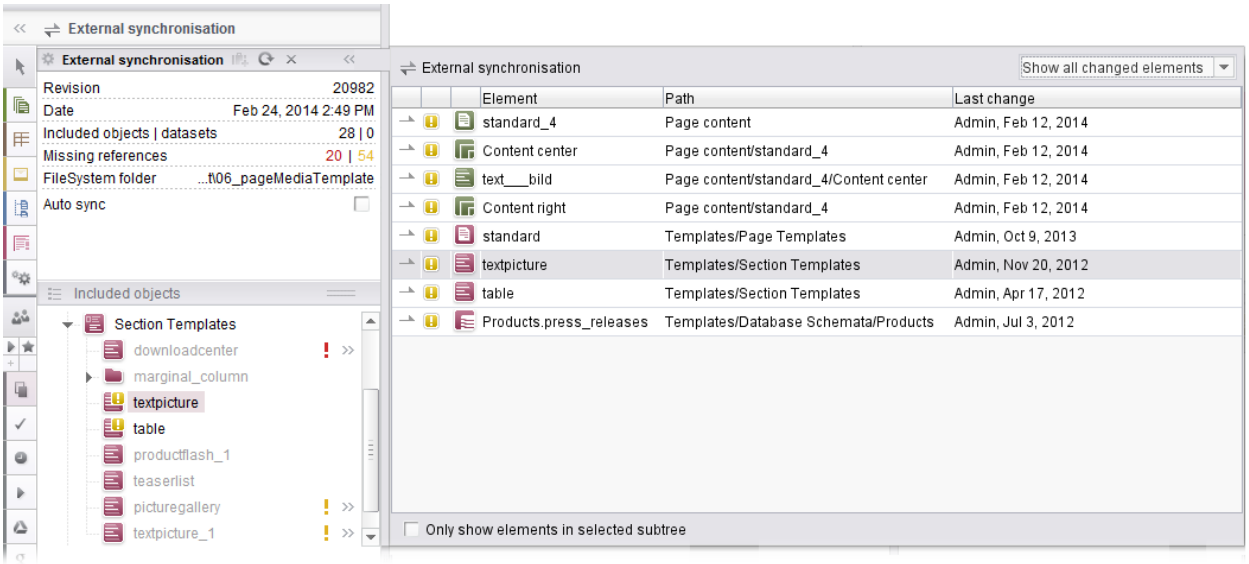


Figure 7-11: Synchronization overview

In addition to

- object type icon,
- name,
- path and
- date, as well as the originator of the last change to the object,

the following information about the objects to be synchronized is displayed in table form in this overview:

Action: The intended action to be carried out during the next synchronization of this object is displayed in the first column. A particular action is pre-assigned to the object depending on its synchronization state (see section 7.2.3.1, page 65). However, this default action can be changed by double-clicking on the appropriate icon in this view. Another way of changing the action is to use the object's context menu (see section 7.2.3.3, page 69).

Export: The next time synchronization is performed using the icon, the FirstSpirit element is exported to the selected file system folder. New elements are always exported. Double-clicking on this icon reverses the direction of synchronization from export to import.

Import and update: The next time synchronization is performed using the icon, the FirstSpirit element is imported from the selected file system folder to the current project. Elements changed externally are always imported. Double-clicking on this icon reverses the direction of synchronization from import to export.

Do not synchronize: The next time synchronization is performed using the icon, the FirstSpirit element is neither exported nor imported. Double-clicking to change the action is not possible.

Conflict: Here a conflict is present, since both the exported object and the internal object have been changed in the project since the last synchronization. In this case, the user must decide which change to apply during the next synchronization process. A default action therefore does not take place (refer to Resolving conflicts in section 7.2.7, page 75). Double-clicking on this icon reverses the direction of synchronization from "Do not synchronize" to import or to export.

Delete: The next time synchronization is performed using the icon, the FirstSpirit element is removed from the selected file system folder. Double-clicking to change the action is not possible.

State: The synchronization state is shown in the second column. (Refer to section 7.2.3.1, page 65.)

In the top area of the flyout menu, the user can filter the displayed list:

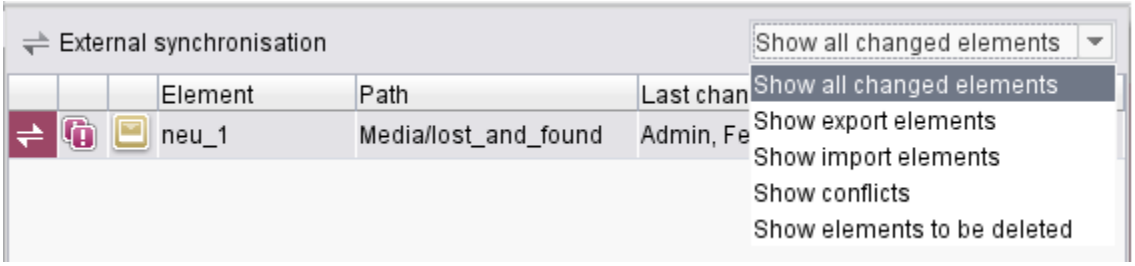


Figure 7-12: Filtering the displayed objects

- **Show all changed elements:** Only elements that will be changed during the next synchronization process are displayed. No differentiation is made between external and local changes. All elements to be imported, exported and deleted as well as all elements that currently have conflicts and cannot be synchronized are displayed.
- **Show export elements:** Only elements that will be exported during the next synchronization process are displayed.
- **Show import elements:** Only elements that will be imported during the next synchronization process are displayed.
- **Show conflicts:** Only elements with conflicts are displayed.
- **Show elements to be deleted:** Only elements that are to be deleted are displayed.

At the bottom of the flyout menu window is an option that can be used to restrict the filtered list to individual subtrees: **Only show elements in selected subtree:** Selecting this checkbox restricts the display of elements to a subtree selected in the "Included objects" area. For instance, if this checkbox is activated and the "Templates" node is selected, only the explicitly included "templates" are displayed, but not the included "content". Of course, nodes that are further down in the hierarchy can also be selected.

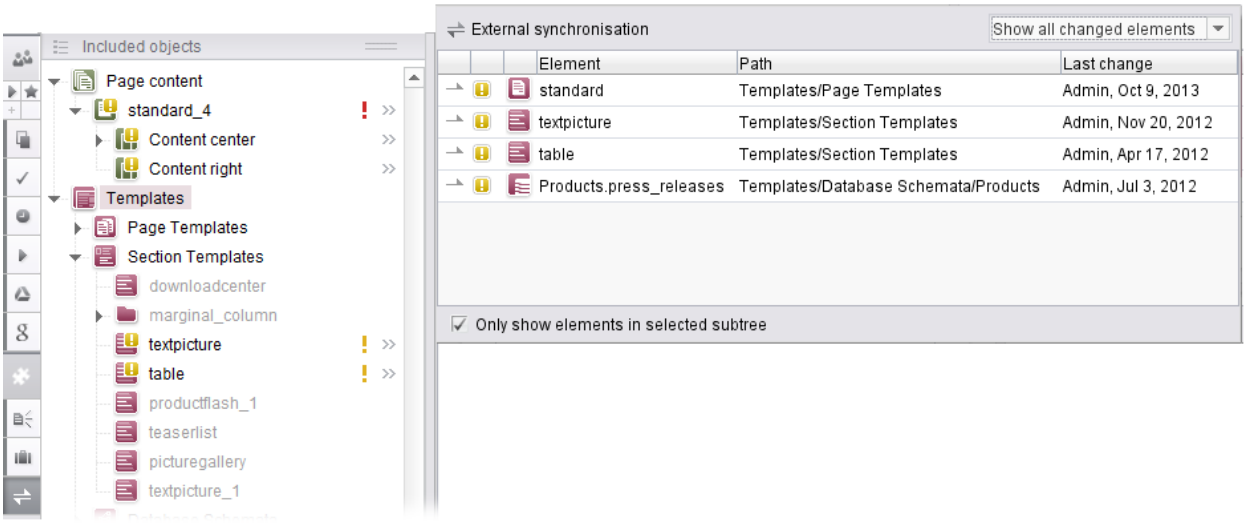


Figure 7-13: Only show elements of a subtree

7.2.7 Resolving conflicts

The "External synchronization" function makes it possible to synchronize individual FirstSpirit objects from various projects in an external directory. This means that an external folder can contain files from multiple FirstSpirit projects and can in turn import

them to other FirstSpirit projects. The synchronized objects/files can be changed both in the individual projects as well as in the external folder. During synchronization, conflicts may occur if, for instance, an object in a synchronization interval has been changed locally in the project as well as externally in the directory. In case of conflict, default actions cannot be specified by the system. This type of conflict must be resolved by the user in order to prevent unintentional overwriting of changes.

The conflict state is indicated by the icon in front of the object in the "Included objects" area. A tooltip for the object shows additional information about the cause of the conflict (see section 7.2.3, page 63).

The conflict can now either be resolved directly in the "Included objects" view (via the context menu; refer to section 7.2.3.3, page 69), or via the "External synchronization" detail view (flyout menu).

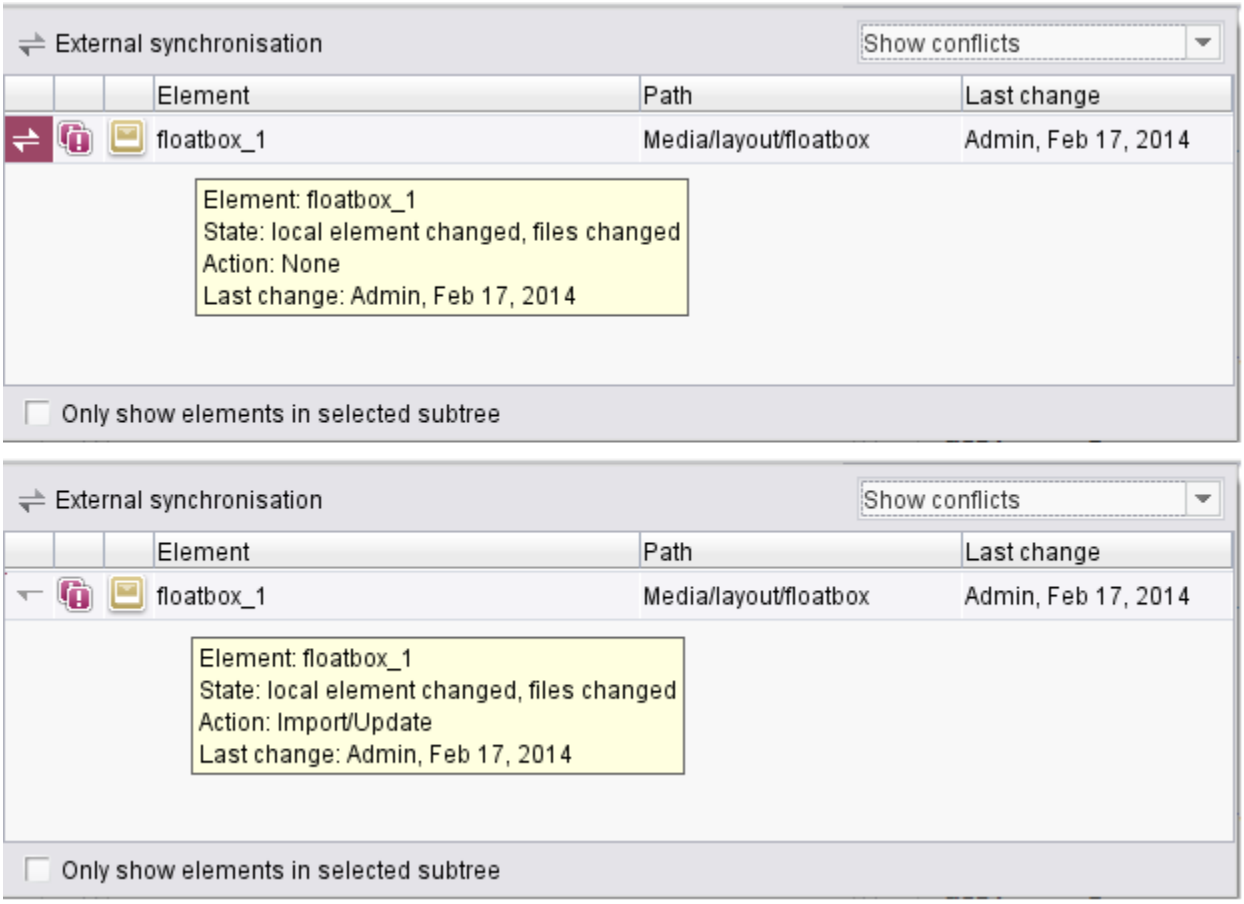


Figure 7-14: Conflict

To display all objects that have conflicts, first select the "Show conflicts" filter (see Figure 7-12 for filter options). Only objects with the conflict state will now appear in the list.

The "Select action" icon indicates to the editor that manual selection is required in order to resolve the conflict. The user can reverse the direction of synchronization to ("export") or to ("import") by double-clicking on the icon. The same toggle process can be performed using the context menu for the element (see section 7.2.3.3, page 69). Manually resolving the conflict will remove the "Conflict" state during the next synchronization process. The local or external changes to the object will then be overwritten as the editor desires.

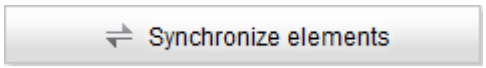
7.2.8 Synchronizing objects

FirstSpirit objects can be synchronized externally in two ways: either via the "Auto sync" entry in the configuration area or manually using the "Synchronize elements" button.

Both options require that an external folder is selected first. This folder forms the basis for the exchange of objects/files with the file system. Elements in this folder can be synchronized from the local FirstSpirit projects. Included objects are listed under "Included objects" in the "External synchronization" area. If objects from a FirstSpirit project are to be synchronized not only externally but the intention is also to use these objects in other FirstSpirit projects as well (by importing them from the external folder), the user must ensure that all dependencies of the objects are also included in the synchronization or that these dependent elements are already present in the target project. The external synchronization displays these references to the user (for the objects to be exported), but synchronization can however be started even if not all references have been met.

The mode for synchronization should not be established until all objects are included. Here the user can choose between automatic or manual modes.

- **Automatic synchronization** is activated using the "Auto sync" checkbox (see section 7.2.1, page 59). All included objects are synchronized without further editing by the user in the defined folder as long as the folder remains open in the "External synchronization" area. Synchronization takes place whenever one of the included objects changes (during saving) and the user's local SiteArchitect is in the foreground. Synchronization is always based on the state of the object and the associated default action (see section 7.2.3.1, page 65). Changing the direction of synchronization or resolving conflicts automatically is not possible in this mode. Dependent objects are not added automatically. Since new objects are synchronized immediately as well, it is recommended to keep the checkbox unchecked until all objects and their dependencies have been captured first.

- 

Synchronize elements; clicking on this button starts a one-time **manual synchronization** of "included objects" (with the option to export changes to the external directory or import external changes to the FirstSpirit project).

  The "Included objects" overview always shows the state of the objects in the external directory. These objects are linked to a revision in the project. In order to

find local or external changes that have been made to objects since the last synchronization, this revision must be updated using the "Reload FileSystem folder" icon prior to manual synchronization (see section 7.2.1, page 59). If external content has changed since the last synchronization, these changes will now be shown.

Prior to manual synchronization, dependencies can be added (see section 7.2.4, page 71 and section 7.2.5, page 72), the default synchronization action can be changed, or objects can be removed (see section 7.2.3.3, page 69). In addition, existing conflicts related to particular objects can be resolved (see section 7.2.7, page 75).

Only one version of an object is synchronized at a time. This means that if an object is exported to an external directory and then reimported (e.g. into a different FirstSpirit project), the version history only receives the corresponding add or import operations, but not the entire version history of the object from the source project.

When synchronizing FirstSpirit project content, objects are changed, deleted and added. Regardless of the action performed, the relevant user permissions (read, edit, create) are required.

7.2.9 Reproducing data in the file system

FirstSpirit objects are stored in the external directory as files where they can also be edited. When exporting FirstSpirit objects to the file system, an attempt is made to reproduce the hierarchy in the project to a folder hierarchy in the file system. Contextual information related to the exported FirstSpirit objects is stored in XML files. The following principles apply; however, there may be exceptions depending on the particular object type:

- A folder is created in the file system for each folder present in FirstSpirit. The path from the object selected in FirstSpirit up to the particular root node is reproduced in the form of folders.
- An XML file is always created for each FirstSpirit element. This file contains information such as the ID, reference name, etc.
- A file is created for each tab.
- Datasets are stored in a file.

7.3 Enhancements in SiteArchitect

The SiteArchitect interface has been extensively overhauled and the design has been updated. There are also some new functions available. New features in the Template Store and Site Store – the most important stores for template developers – are presented below. For new features in the other stores, see section 6.2 starting from page 38.

7.3.1 Site Store

The order of the functions is rearranged at the **menu level**.

The process for creating and managing **page groups** ("Page Groups" tab) has been simplified in version 5.1.

On this tab, all page references at this menu level which have not yet been placed in a page group are listed under "Ungrouped pages":

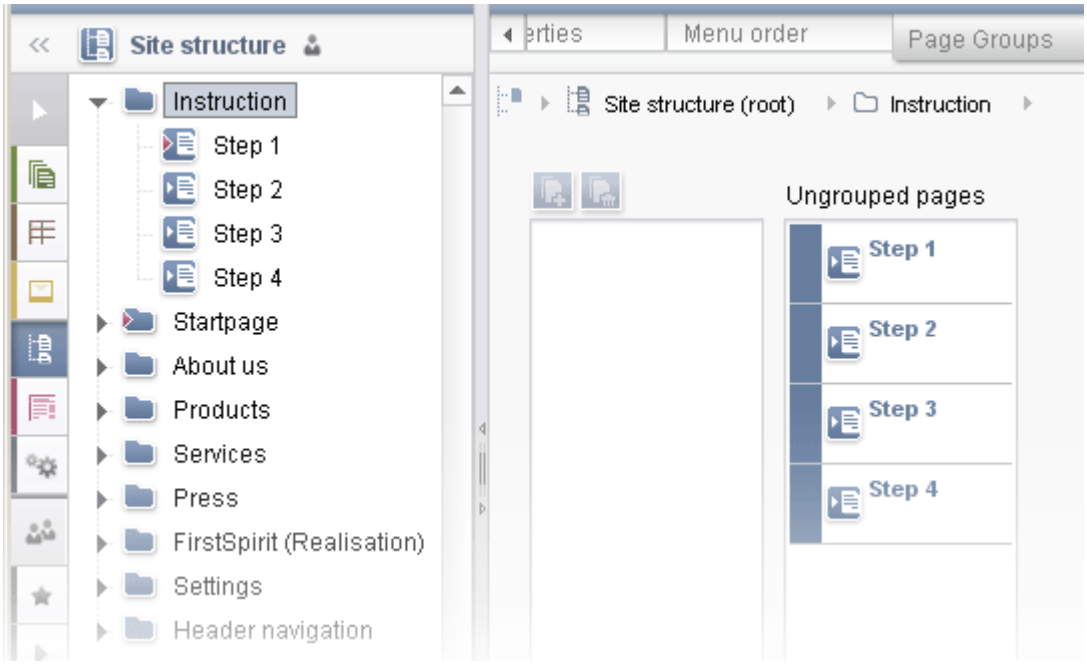


Figure 7-15: Menu level – "Page groups" tab

Create new group: Clicking on this icon opens a window in which the user can enter a name for the new page group. Once the input is confirmed with "OK", the new page group is added to the left-hand column on the tab.

Delete group: Clicking this icon deletes the selected page group.

The page references in the "Ungrouped pages" area which belong in the page group are moved to the page group using drag-and-drop. Several page references can be selected at the same time by pressing the mouse button and <Ctrl> or <Shift>. The user can change the position of the page reference by moving it with the mouse. Pages can also be removed from a page group using drag-and-drop.

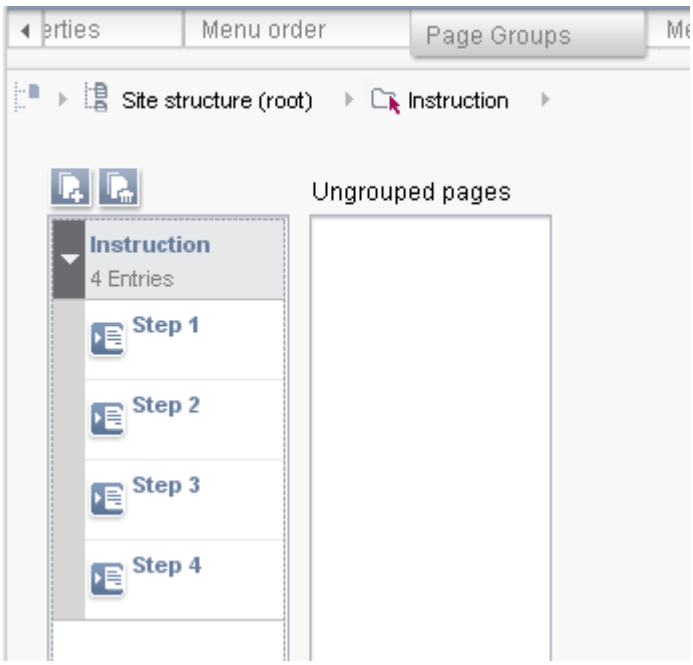


Figure 7-16: Page group with entries

At the **page reference level**, the "New entry" and "Delete entry" functions for defining the points in time for "validity period" are moved from a context menu to icons.

7.3.2 Template Store

There is no longer a separate tab ("Preview") for selecting and displaying a preview image in version 5.1. The preview image can now be conveniently selected from the "Properties" tab via the  icon in the "Preview image" area and closed again with  .

Line numbering can be shown or hidden using the  icon in the top right-hand corner of the workspace for templates.

The previous "Search in templates" function ("Search" menu) has now been replaced by the global search and the search icons on the tabs of templates.

 Edit default values ("Form" tab): This icon is used to open the dialog for editing default values.

 Search (keyboard shortcut <Ctrl> + F): Opens a window in which the user can search for text.

 Replace (keyboard shortcut <Ctrl> + R): Opens a window in which the user can search for and replace text.

 Next occurrence (keyboard shortcut <F3>): Jumps to the next search result.

 Previous occurrence (keyboard shortcut <Shift> + <F3>): Jumps to the last search result.

 Undo (keyboard shortcut <Ctrl> + Z): Undoes the last change.

 Redo (keyboard shortcut <Ctrl> + Y): Restores changes that have been undone.

7.3.2.1 Syntax highlighting

Syntax highlighting in the Template Store has been enhanced for improved readability. Opening and closing tags of instructions were already highlighted in presentation channel tabs in the past. Now even more occurrences of highlighted FirstSpirit code, such as expressions, variables, etc. are highlighted as well.

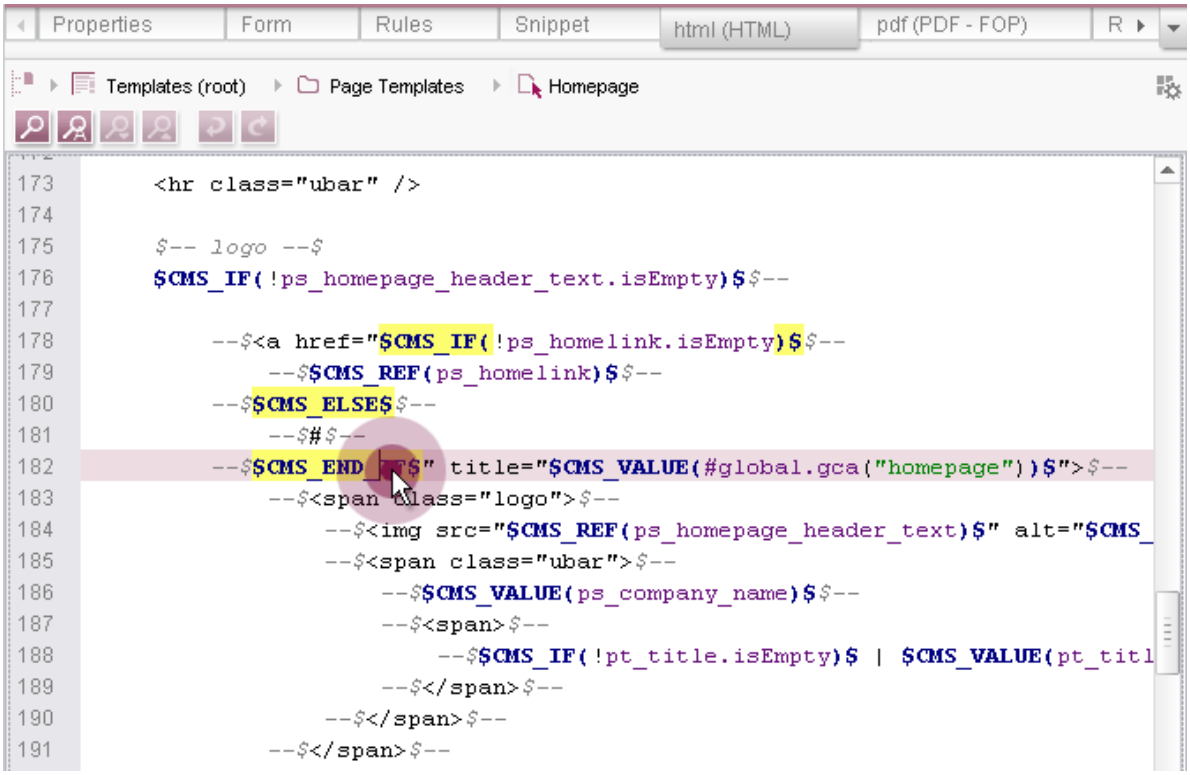


Figure 7-17: Highlighting associated tags

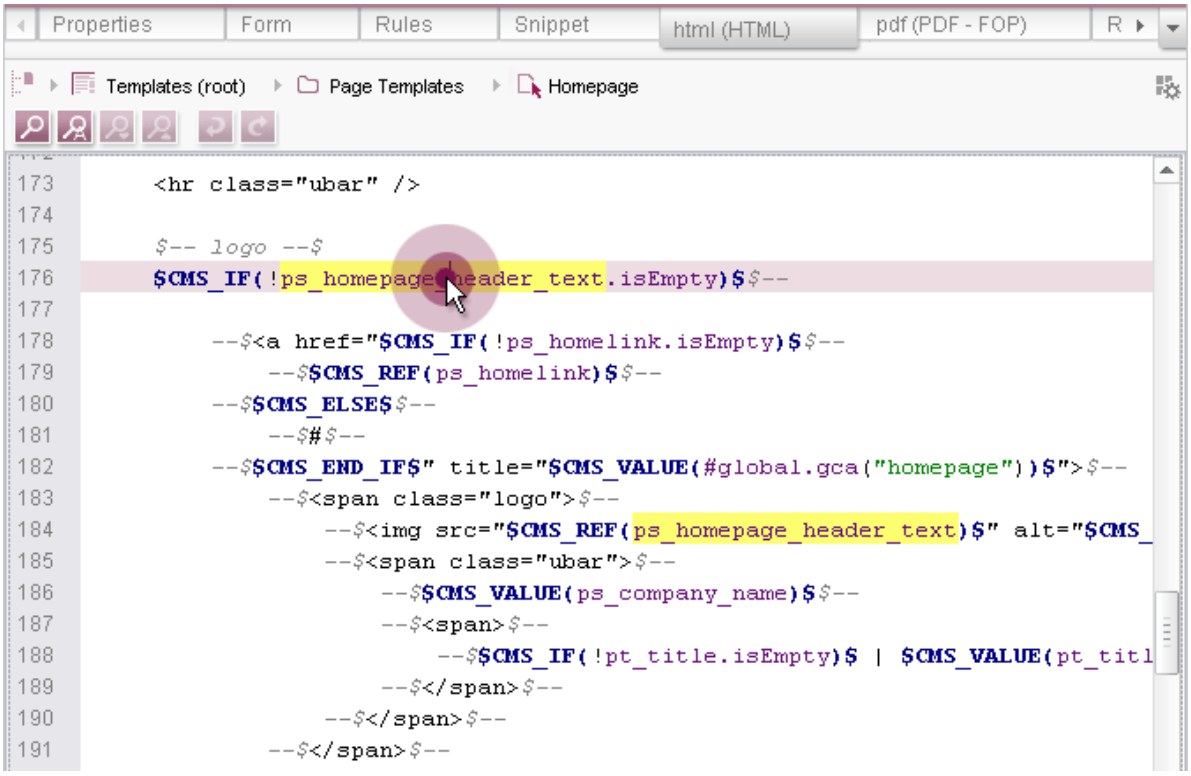


Figure 7-18: Highlighting multiple occurrences of a variable

7.3.2.2 Page templates

Content areas for page templates and section restrictions are now defined on the "Content areas" subtab on the "Properties" tab. The interface has a more modern appearance than previous versions but the functions remain the same:

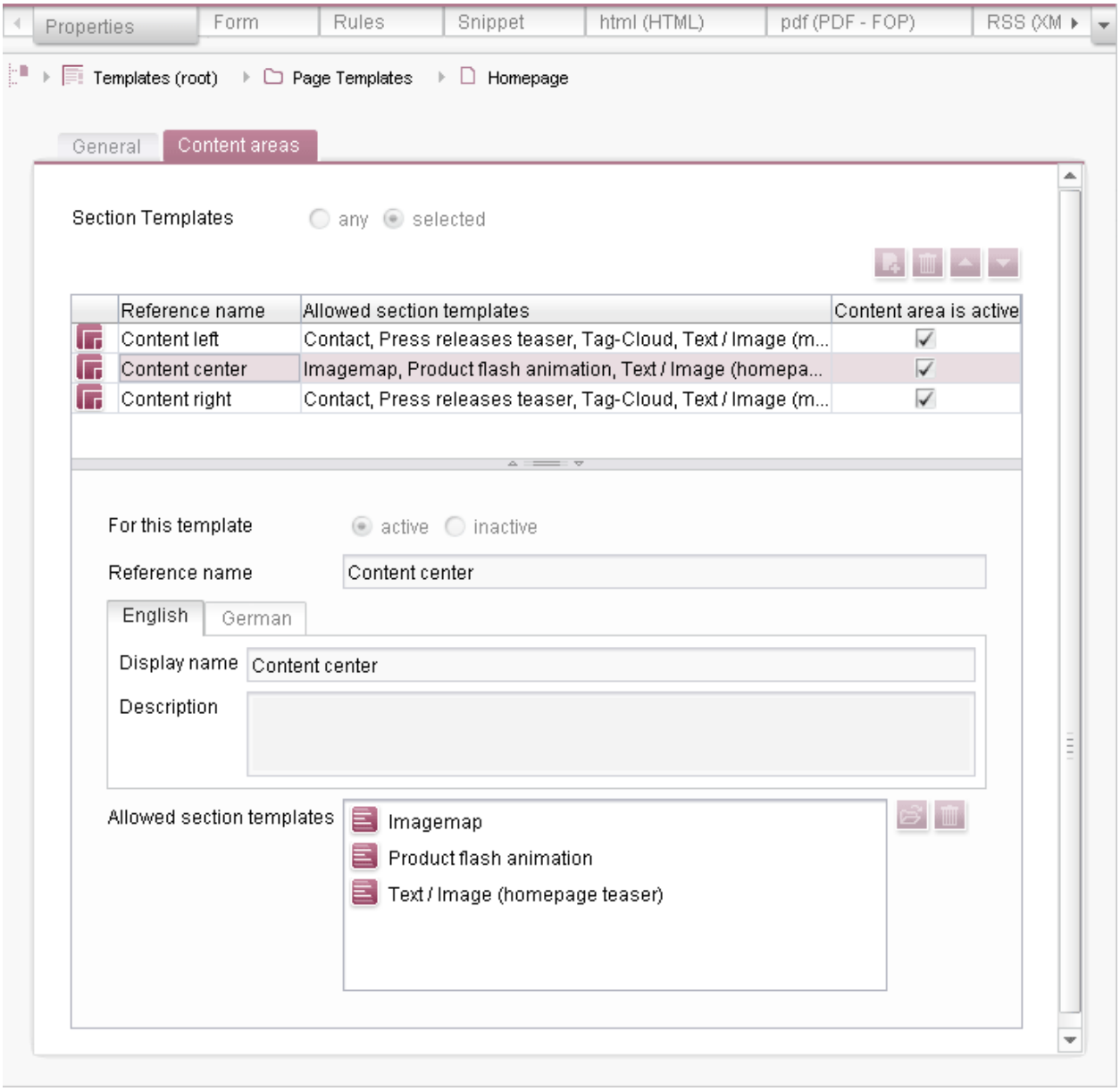


Figure 7-19: Content areas of a page template

Refer also to the section on "Templates (Basics)"/"Structure of templates"/"Page templates" in the FirstSpirit online documentation for more information.

7.3.2.3 Table format templates

If the number of rows and/or columns are to be specified that can or must have an inline table based on the affected table format template, the relevant "limited" checkboxes in the "Table size" area now need to be activated. Fields will then be displayed in which

the minimum and maximum number of rows/columns can be specified.

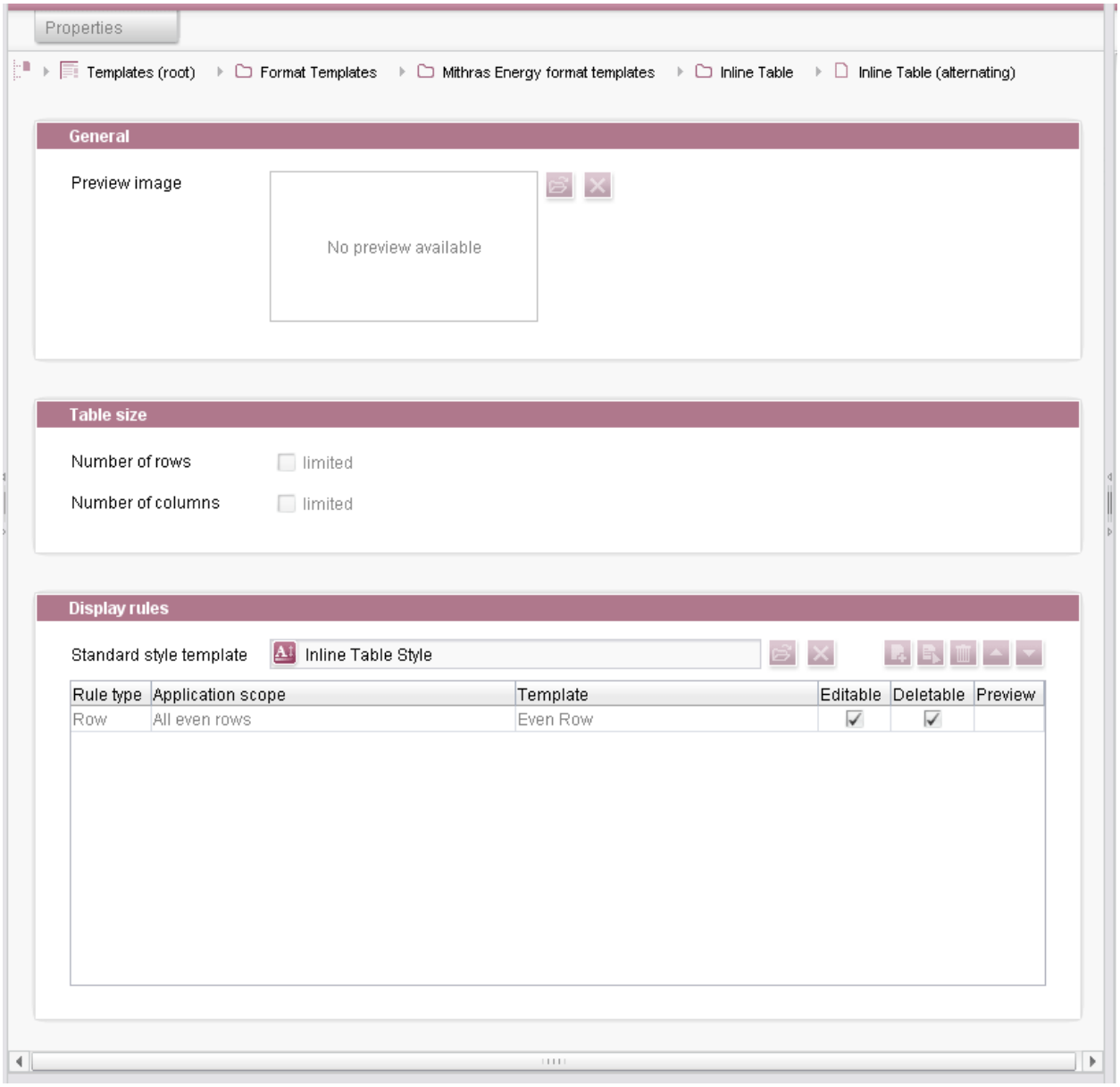


Figure 7-20: Table format template

If the checkboxes are not activated or the specified default setting (see Figure 7-20) is not used, the number of rows/columns will not be limited.

7.3.2.4 Scripts

The interface in the scripts area, particularly the "Properties" tab, has also been updated but the functions remain the same.

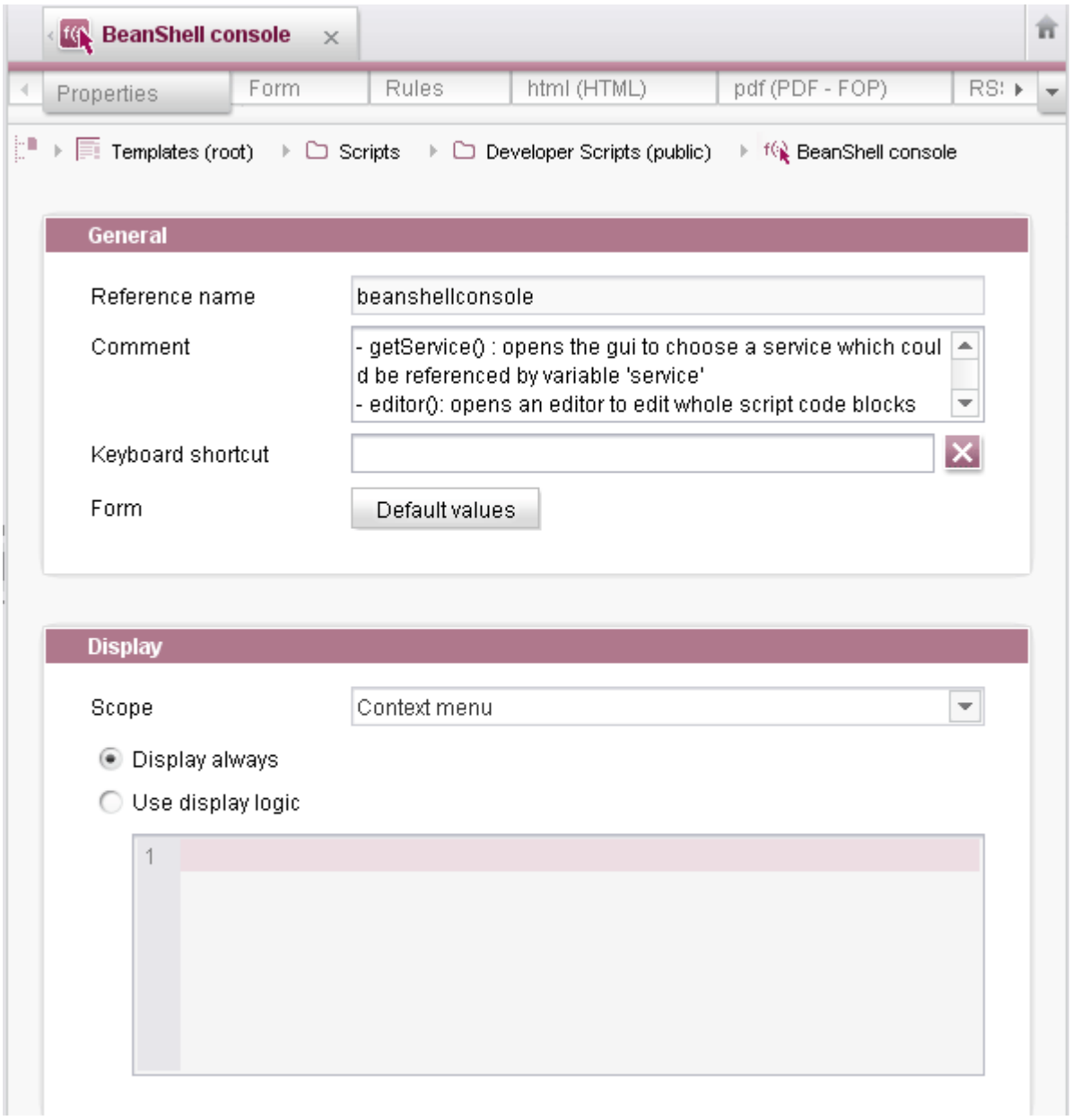


Figure 7-21: Scripts – "Properties" tab

The script type, which was previously selected via the "Script type" drop-down list, is now selected from the "Scope" drop-down list in the "Display" area. If the script is to

appear in the "Extras" menu and on the entry page (see section 6.2.2 page 40), the "Menu and entry page" option must be selected. Previously, this was done by selecting the "Menu" script type and activating the "Use on entry page" checkbox.

Where the display logic was previously activated via the "Always active" checkbox in the "Display logic" area, this is now done via the "Display always" and "Use display logic" options: To have the script always displayed in the selected scope, the "Display always" option must be set. To activate the stored display logic, the "Use display logic" option must be set.

7.3.2.5 The database schema editor

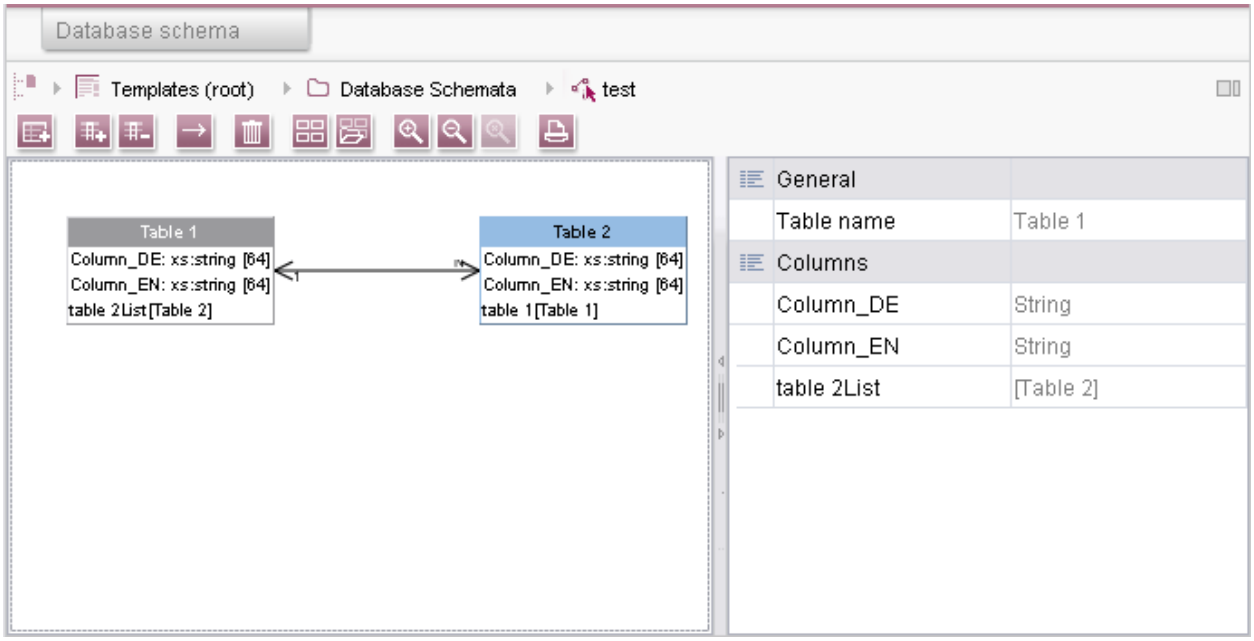


Figure 7-22: The database schema editor

Tables, columns, and foreign key relationships can be created and configured in the database schema editor as before via the icons or the context menu (see "Templates (Basics)"/"Structure of templates"/"Database schemata"/"Schema editor" in the *FirstSpirit online documentation*. The attributes of the individual elements can now also be displayed in a property table. The view can be modified via the icon in the top right-hand corner of the workspace. An alignment grid can also be activated or deactivated here, allowing the elements to be arranged clearly. The property table can only be enlarged to the left by moving the divider. The "Hide system columns" option replaces the previous "Only show usable attributes" function.

The options in the editor remain the same as in previous versions.

7.3.2.6 Table templates

The configuration options for table templates remain the same as in previous versions. Only the order of the tabs has changed. The arrangement of some of the functions and some of the wording has also changed. For example, the column titles on the "Mapping" tab have changed:

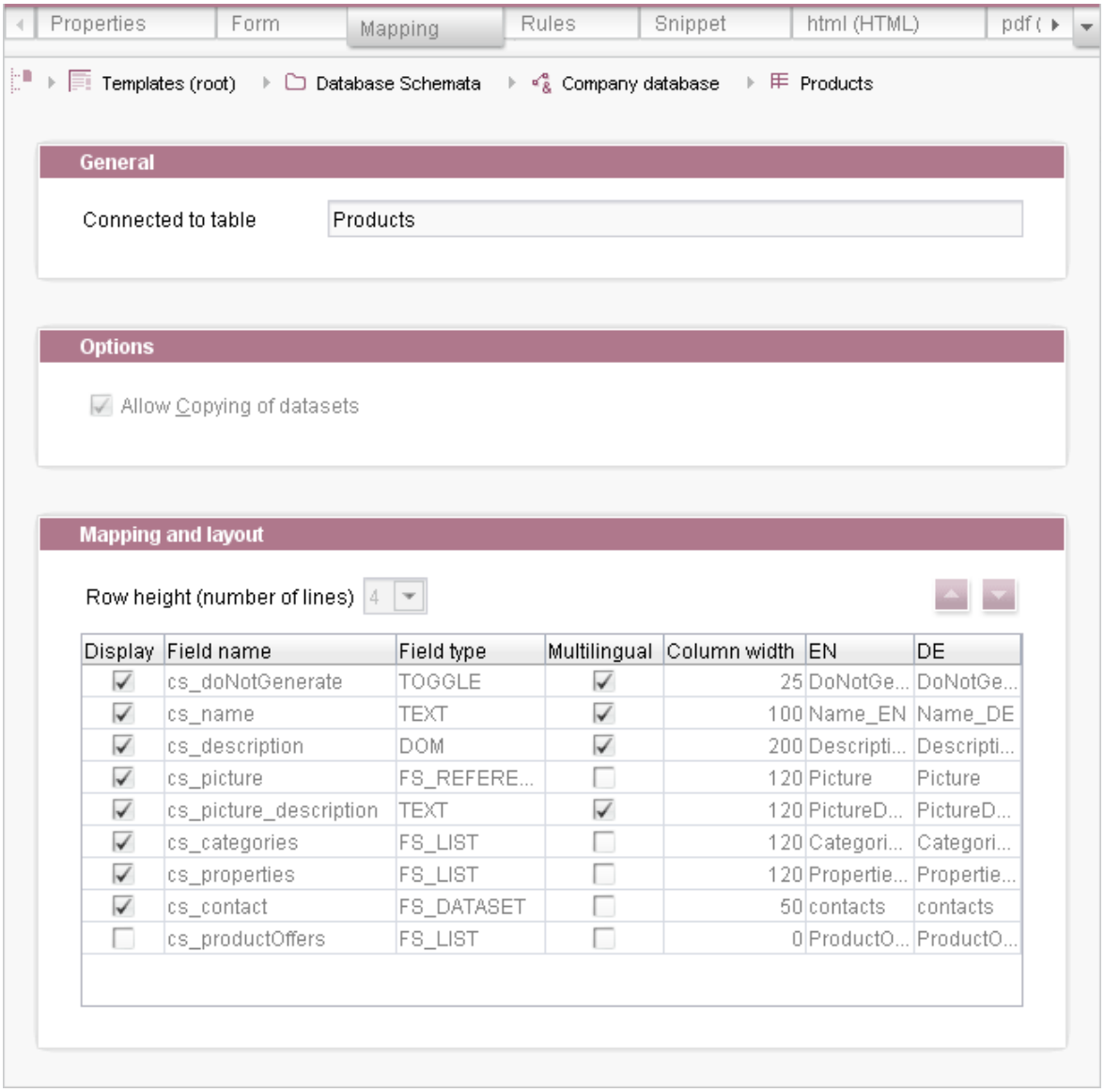


Figure 7-23: Table template – "Mapping" tab

A tooltip provides more information about the relevant column. It is no longer possible to change the width of all columns. The "Mapping" tab and the relevant field are highlighted in color when they contain invalid entries.

7.3.2.7 Queries

The configuration options for queries remain the same as in previous versions. Only the arrangement of some of the functions and some of the wording has changed, e.g. on the **Conditions** tab:

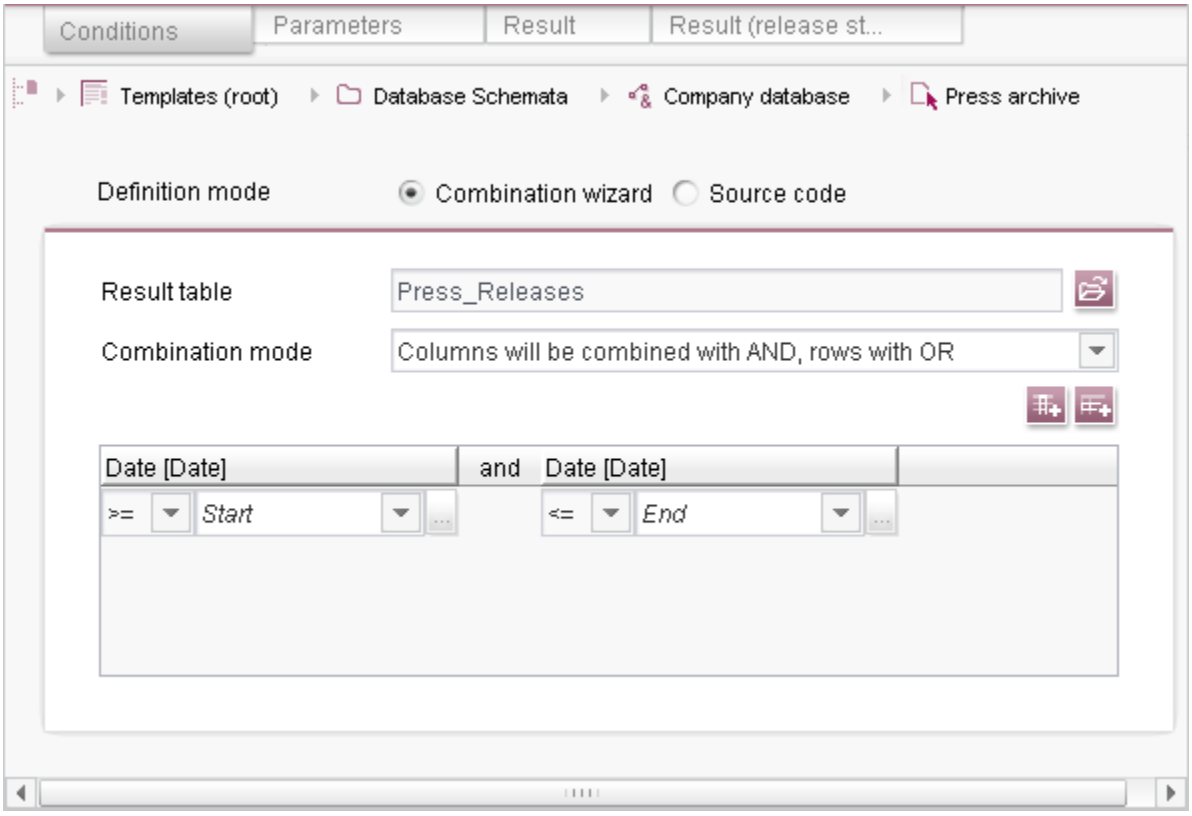


Figure 7-24: Query – "Conditions" tab, "Combination wizard" mode

The combination wizard (previously "Wizard mode") is activated by default. To edit the source text, the "Source code" radio button must be activated.

 This icon is used to select the desired table.

 This icon is used to select the desired column (previously "Add restriction").

 This icon is used to add additional rows.

Values can now be set "inline" on the **Parameters** tab; previously, a pop-up window opened for the user to enter/select values.

The detail view for datasets on the **Result** and **Result (Release State)** tabs, which used to open when the user double-clicked on a dataset, no longer exists.

7.3.2.8 The workflow editor

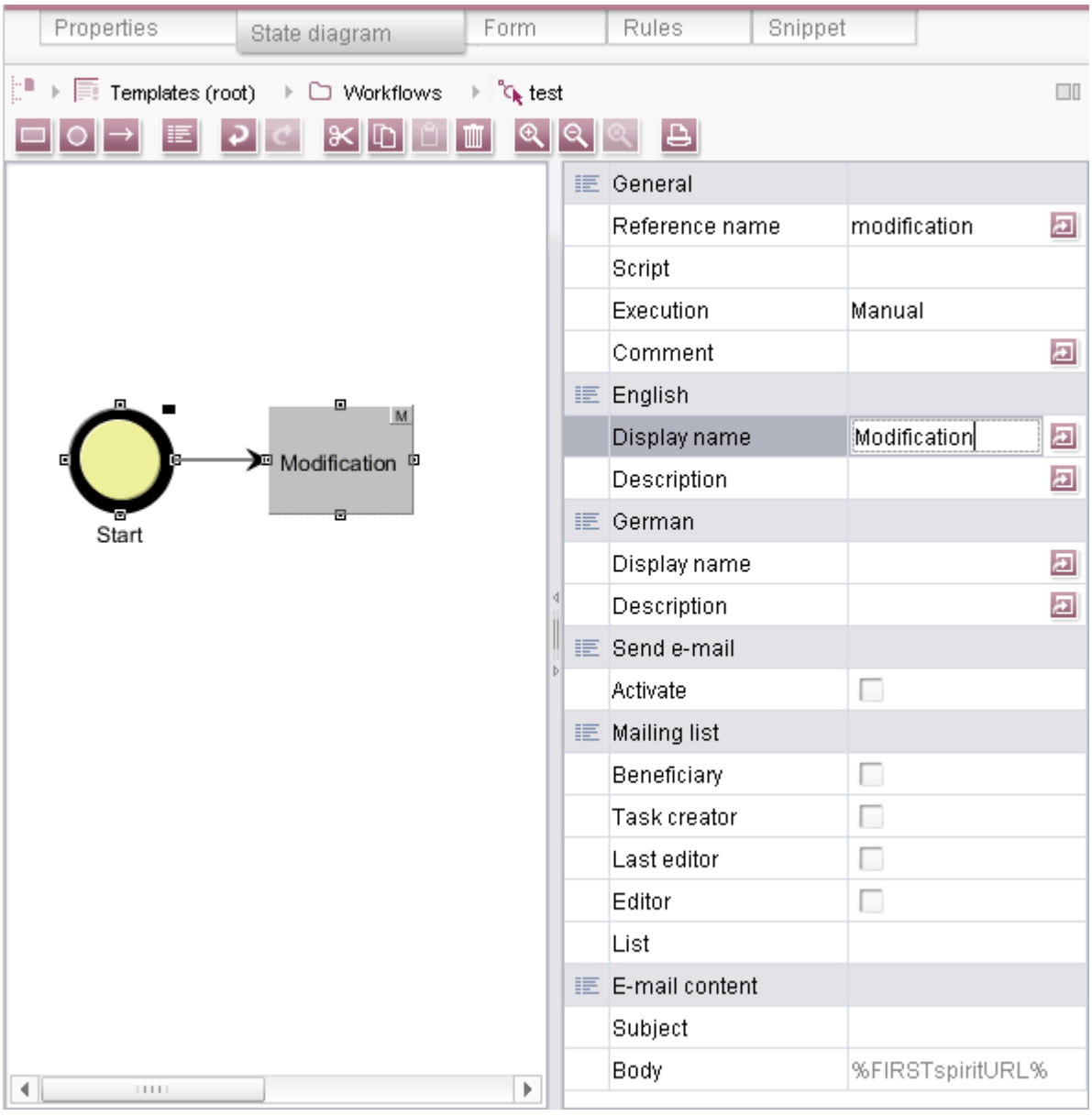


Figure 7-25: The workflow editor

The individual elements of the workflows can be added and configured as before on the "State Diagram" tab via the icons or the context menu (see "Templates (Basics)"/Structure of templates"/"Workflows" in the *FirstSpirit online documentation*). Configuration can now be carried out not just in a dialog (by double-clicking on an element), but inline in a **property table** as well. The view can be modified via the icon in the top right-hand corner of the workspace. An alignment grid can also be activated or deactivated here, allowing the elements to be arranged clearly. The property table can only be enlarged to the left by moving the divider.

The options in the editor remain the same as in previous versions; the only change is that the "Unique name" field is now called "Reference name".

To enter text (including color values), the user can click in the fields and type the text directly into the field. Time values ("Dwelling period" field) can only be entered using the arrow icons.

This icon is used to open a larger window for entering text.

This icon is used to select users and groups in activities, states, and transitions. The following dialog opens:

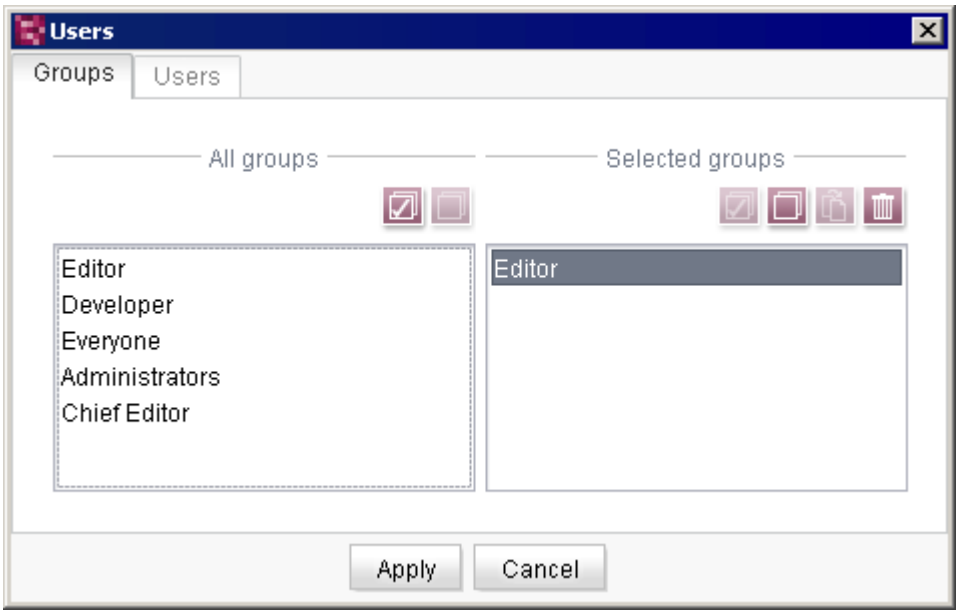


Figure 7-26: Selecting groups or users

This icon is used to select all groups/users in the relevant column.

This icon is used to select all groups/users in the relevant column.

This icon is used to transfer the groups/users highlighted in the left-hand column to the right-hand column. The user then clicks "Apply" to select these groups/users.

This icon is used to remove a selected group or user from the right-hand column.

This icon is used to select the desired color coding for the current workflow state. The following dialog opens:

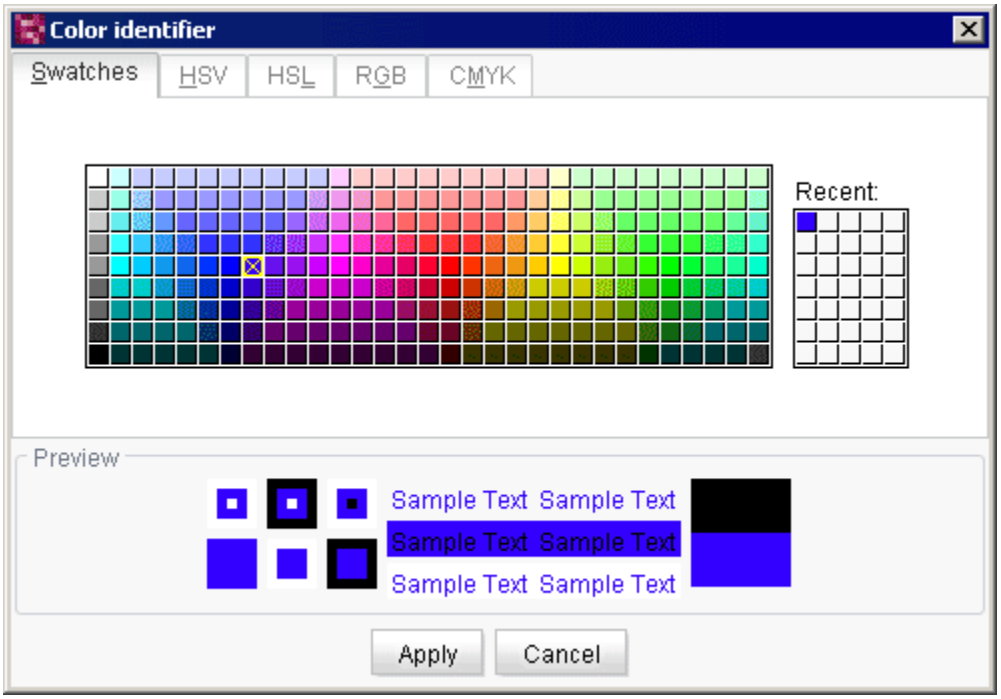


Figure 7-27: Color selection

Colors can be chosen according to different color models using the different tabs.

The other workflow tabs remain the same, apart from the order in which they appear.

7.3.3 Reports in SiteArchitect

In FirstSpirit version 5.1, the reports that users will know from ContentCreator can also be used in SiteArchitect. (See also section 6.2.3 page 41.)

Figure 7-28 shows the integration of a report in the left-hand navigation area in SiteArchitect. The editor can enter search terms in a search box in SiteArchitect and limit the number of hits to specific file types (1) – in this case, web content, images, and videos. The search is handled by the Google Search API and can be configured for a specific project. SiteArchitect then displays the results in the form of a uniform, seamless list (2) integrated in the UI. The results can be copied from the report to the editing area of SiteArchitect (3) via drag-and-drop:

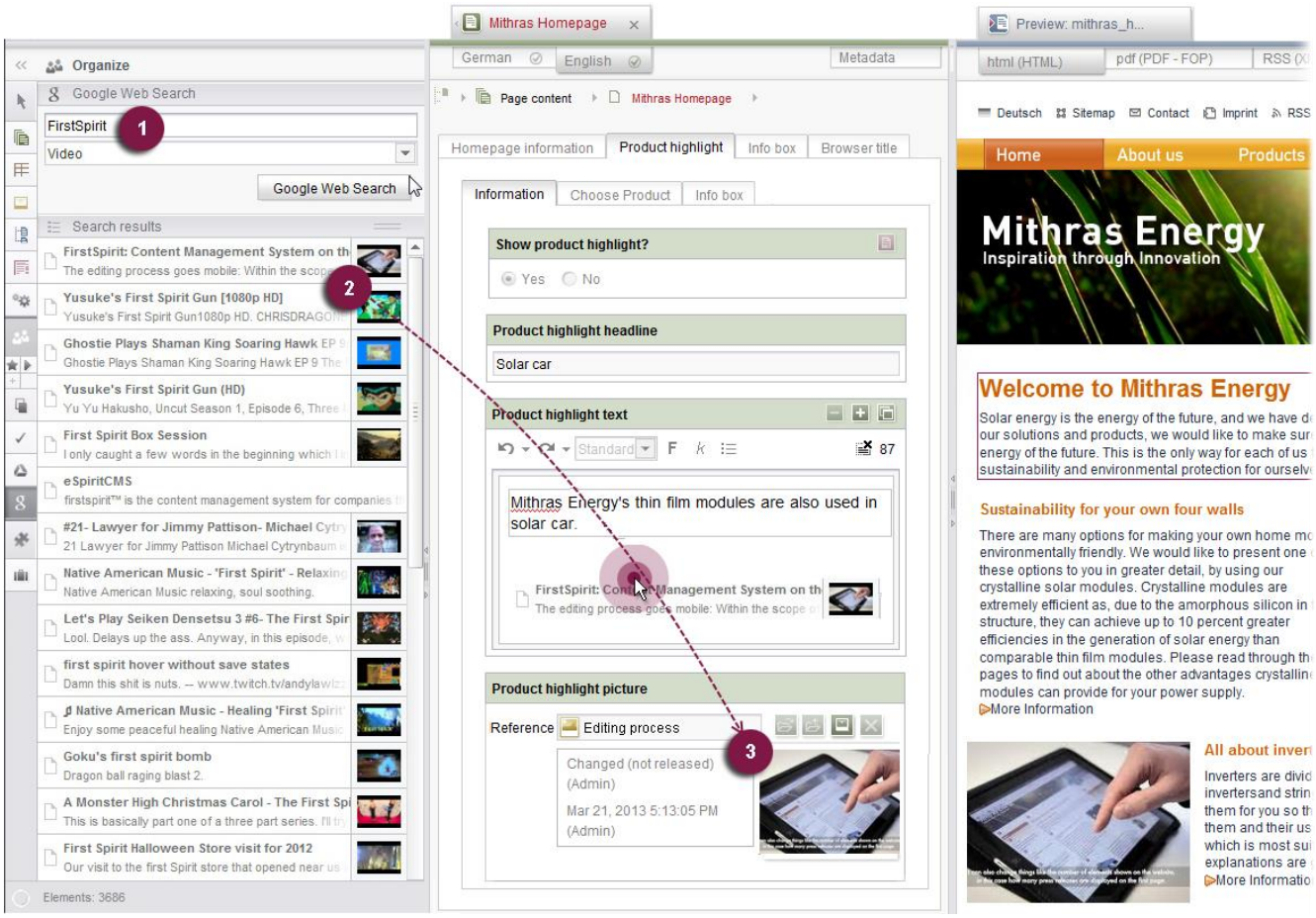


Figure 7-28: Drag-and-drop from the integrated Google Web search

The new interface `de.espirit.firstspirit.client.plugin.ReportPlugin` has been implemented in the FirstSpirit Developer API for this purpose (see also section 7.5.2 page 107).

For detailed information on implementing reports in SiteArchitect, see "Plug-in development" in the FirstSpirit online documentation and the documentation about the FirstSpirit AppCenter.

7.3.4 Snippets for datasets

Snippets of table templates are now also used for the overview in the left-hand client column as well as the path displayed in the workspace (see Figure 6-23). If no definitions are given for the "Thumbnail", "Label", and "Extract" fields, the name of the relevant data source and the dataset ID are displayed in the overview, e.g.

```
Contacts#2240
```

For detailed information on the definition of snippets, see FirstSpirit online documentation, "Template development"/"Snippets" and documentation about the "FirstSpirit AppCenter".

7.3.5 Reference graph: Enhancement of dataset referencing

The reference graph is used to find dependencies within a project and is therefore an essential component of complex functions such as server-side releases. Moreover, when deleting FirstSpirit objects, it is also possible to redirect the existing object references to other FirstSpirit objects in order to avoid still having invalid references to deleted objects, for instance.

The visualization of object referencing can be requested at the respective node in the tree structure using the “Extras” / “Display dependencies” context menu or the <CTRL> + <R> keyboard shortcut (only available to project administrators, however). In addition, the FirstSpirit API can also be used to access incoming and outgoing references, such as via the `getIncomingReferences` method.

The reference graph in FirstSpirit version 5.1 now takes into account more incoming references to datasets, such as those from

- **queries** (Template Store)
  The queries reference graph displays the dataset that is the result of the query.

- **filtered data sources** (Template Store)
  The filtered data sources reference graph displays the dataset upon which the data source is filtered.
- **content projections** (Site Store)
  (page references based on pages with an integrated data source): The content projections reference graph displays the dataset that was selected in the “Select a query” area.

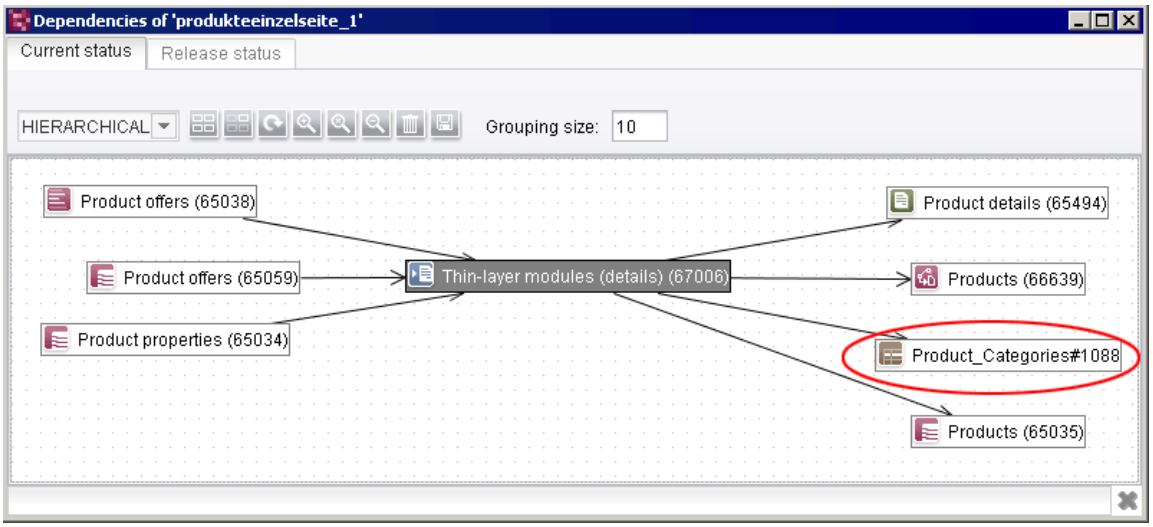


Figure 7-29: Dependencies of a content projection

These elements must refer to only to the dataset that is determined based on the dataset ID by using, for instance, a query such as the following:

```
<QUERY entityType="Products">
    <FILTERPARAM parameter="category" datatype="java.lang.Integer"
        value="1090"/>
    <EQ attribute="Categories_List.fs_id" parameter="category"/>
    <ORDERCRITERIA attribute="Name_%lang%"/>
</QUERY>
```

Access via the API is possible using the “getIncomingReferences” method, in this case the one for the `Schema` interface (Package: `de.espirit.firstspirit.access.store.templatestore`).

7.3.6 Updates to the input component CMS_INPUT_IMAGEMAP

The input component CMS_INPUT_IMAGEMAP has been updated in FirstSpirit 5.1.

When this input component is used in SiteArchitect, the new *simpleMode* parameter limits the frame shape to a rectangle, while *resolution* specifies the resolution to be used for the selected background image. For more information, see *FirstSpirit online documentation*, "Template development"/"Forms"/"Input components"/"IMAGEMAP".

For information on functional changes for the editor, see also section 6.2.10, page 53.

7.4 Enhancements concerning ContentCreator

7.4.1 Logging exceptions

In version 5.1, exceptions which occur in ContentCreator during operation are now also logged by the web server. This makes it easier to troubleshoot and analyze errors which occur in ContentCreator. Each log message contains the following text:

```
WebEdit client error occurred:
```

7.4.2 Support for CMS_INPUT_IMAGEMAP

The input component CMS_INPUT_IMAGEMAP can be used to embed links at various locations in a selected background image. First, a rectangular frame is created for each link. The size and shape of the frame can be adjusted to suit the desired location in the background image. The link can then be embedded in this frame (or "mouse-sensitive area").

With the release of FirstSpirit version 5.1, this input component is now also supported in ContentCreator. See also section 6.1.2, page 20 for more information.

For information regarding the parameters for the input component, see *FirstSpirit online documentation*, "Template development"/"Forms"/"Input components"/"IMAGEMAP".

The input component has also been updated for use in SiteArchitect. See section 7.3.6, page 99 for more information.

7.4.3 Making it easier to work with links

Whereas links could previously only be created in the rich text editor (CMS_INPUT_DOM) and the rich text editor for tables (CMS_INPUT_DOMTABLE) via the  icon, they can now also be generated by using drag-and-drop to move elements to the editor, depending on the project configuration. This text is automatically used as the link text when the settings are configured accordingly by the template developer.

To do this, the input component which is to be used as the drop zone in the DOM editor or DOM table in ContentCreator and under whose identifier the drop zone is to be saved must be selected from the "Drop Editor" combo box on the "Properties" tab under "Form variables assignment". All identifiers for the input components defined on the "Form" tab that are of the type

- FS_BUTTON
- FS_REFERENCE
- FS_LIST
- FS_DATASET

are displayed here.

When used in ContentCreator, the editor can choose from all link templates in which the input component type selected here is compatible with the object being dropped and which may be used in the relevant DOM editor (tag *LINKEDITORS*). If only one link template is compatible, the link is automatically created using this template.

If no corresponding input component has been defined, the selection remains empty (*<not assigned>*). If *<not assigned>* is selected, dropping in ContentCreator is not possible with this link template. Refer also to section 6.1.5, page 27 and the paragraph on "Creating links using drag-and-drop".

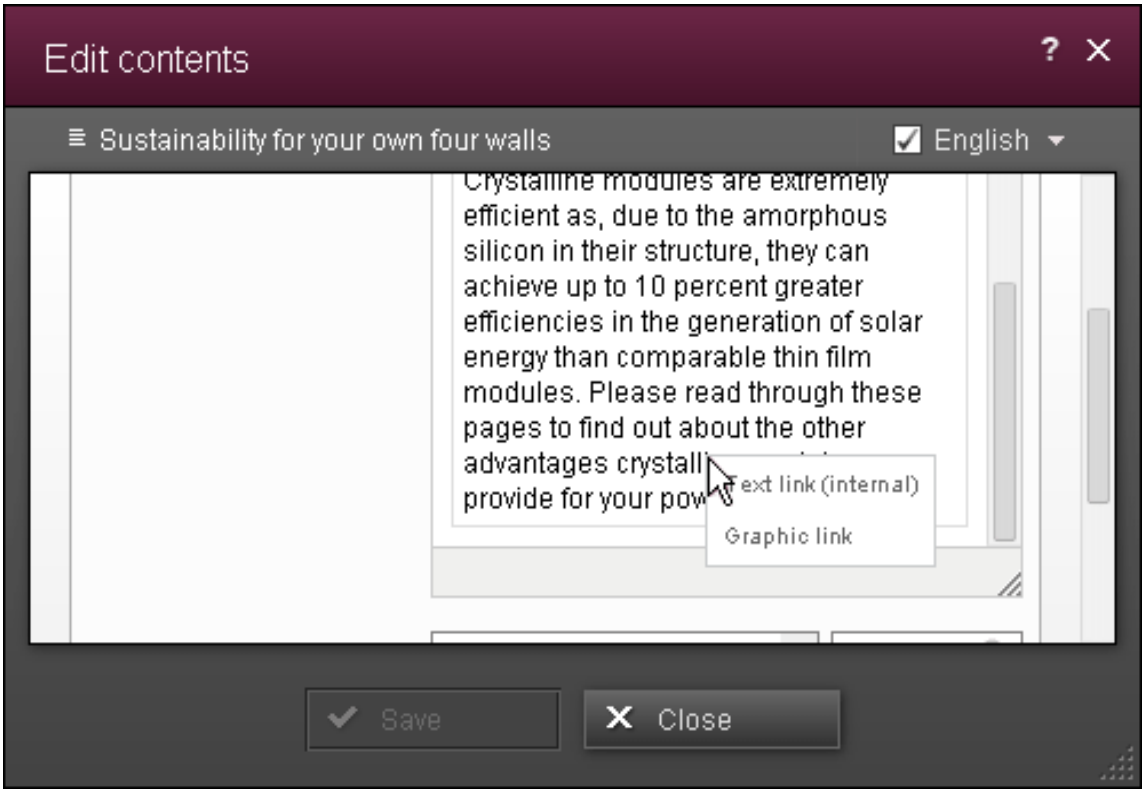


Figure 6-10: Different link creation types

With FirstSpirit version 5.1, depending on the template developer's specifications, users can click on a link to display additional information in a tooltip. See also section 6.1.5 page 27 and the paragraph on "Links with tooltip". A snippet is defined in a similar way to other template types (see *FirstSpirit online documentation*, "Template development"/"Snippets").

For internal links, it is advisable to use an image, the display name, and a passage of text from the referenced page for the tooltip display, e.g.

"Thumbnail" field (displays an image that is saved in the input component with the identifier *pt_highlightPicture*):

```
lt_reference.get.getPage().getFormData().get(#global.language,
"pt_highlightPicture").get()
```

"Label" field:

```
lt_reference.get.getDisplayName(#global.language)
```

"Extract" field (displays the text (limited to 65 characters) that is saved in the input component with the identifier *st_text*)**:**

```
truncate(lt_reference.get.getPage().getFormData().get(#global.languag
e, "st_text").get(), 65)
```

This configuration is also used for the display in CMS_INPUT_IMAGEMAP (see Figure 6-2).

7.4.4 Moving menu items on the preview page

Previously, pages or menu items in ContentCreator could be moved via a dialog which visualizes the project navigation ("Contents" menu/"Edit navigation").

With FirstSpirit version 5.1, it is now possible to move pages or menu items directly on the preview page. See also Figure 6-3 for more information.

To do this, the surrounding HTML element of the relevant navigation element must be include an `editorId()` call in the navigation function.

Example of use with *li* elements (lists)

```
<CMS_ARRAY_ELEMENT>
    <![CDATA[<li$CMS_VALUE(editorId(element:#nav.ref))$>]]>
</CMS_ARRAY_ELEMENT>
```

Example of use with *a* elements (hyperlinks)

```
<CMS_ARRAY_ELEMENT>
       <![CDATA[<a$CMS_VALUE(editorId(element:#nav.ref))$
                href="$CMS_REF(#nav.ref)$">
                      $CMS_VALUE(#nav.label.convert2)$</a>]]>
</CMS_ARRAY_ELEMENT>
```

In some cases, the move function cannot be displayed correctly with icons, e.g. when CSS or JavaScript is used for navigation.

For more information on the editorId function, see "FirstSpirit online documentation"/"Template development"/"Content Highlighting and EasyEdit"/"Use in a project" and "Template development"/"Template syntax"/"Functions"/"In instructions"/"editorId"; for information on the navigation function, see "Template

development"/"Template syntax"/"Functions"/"In header"/"Navigation".

7.4.5 Configuring previews for ContentCreator

As Internet-enabled mobile devices such as notebooks, tablet PCs, and smartphones become more and more widespread, website designs need to be more and more flexible, with content which can be displayed perfectly on different display geometries and in different resolutions. Consequently, FirstSpirit now makes it easy for editors to check the display and navigation of website content with a variety of display sizes in the integrated preview. It also allows content, layouts, and images to be perfectly adapted to suit the output device concerned.

- The configuration for simulating different **display sizes** in the ContentCreator is handled using the settings in the ServerManager project properties; see also section 8.3.1, page 118.
- A **timeline** for displaying the development of a page over time is provided in the ContentCreator by default (see Figure 6-14). (See section 7.4.5.1, page 103.)
- The configuration for simulation of **different aspects** (e.g. preview from the perspective of different users) can be made in the relevant project via a page template that provides the desired input components in the ContentCreator. Also refer to section 8.3.2, page 121 and section 7.4.5.2, page 105 for more information.

For more API functions (`de.espirit.firstspirit.webedit.client.api` package), also refer to section 7.5.2, page 107. For the use of the interface `Preview` see also FirstSpirit online documentation, Chapter "Plug-In Development" / "Implementation and Deployment" / "Using FirstSpirit APIs" / "JavaScript API (ContentCreator)" / "Preview".

7.4.5.1 Timed previews

The editor can use the timeline to select a desired point in time. The display of the current page at this point in time is simulated.

For example, the Validity periods of sections is evaluated for this purpose:
Validity periods for sections can be set in SiteArchitect using the function "Edit validity period" in the context menu of sections (see "Editing the validity period" in the *FirstSpirit SiteArchitect documentation*). If a validity period is defined, the affected section is displayed on the page for the times within this period.

Validity periods for sections can be set in the ContentCreator using the  icon. To do this, the ZIP file containing the ContentCreator examples (*FirstSpirit-ContentCreator-5-Plugin-Examples.zip* under "Plug-in development"/"Examples" in the FirstSpirit online documentation) must be installed.

The points in time selected by the editor can be output as follows:

- **Accessing future points in time**
  If the point in time currently set in the timeline is in the future, it can be accessed using the *#startdate* system object. Documentation about *#startdate* can be found in the FirstSpirit online documentation under "Template development"/"Template syntax"/"System objects"/"#startdate". *#global.startTime* is a synonym for *#startdate*.

- **Accessing past points in time**
  If the point in time currently set in the timeline is in the past, it can be accessed using the runtime variable

```
fs.preview.#time
```

- **Evaluation during runtime (JavaScript)**
  The selected point in time can be output by means of:

```
WE_API.Preview.getTimeParameter()
```

7.4.5.2 Project-specific previews

- **Forms**
  Using input components defined with the page template, parameters can be queried for the project-specific preview view in ContentCreator. The following input components are available for this purpose:
  - CMS_INPUT_CHECKBOX
  - CMS_INPUT_COMBOBOX
  - CMS_INPUT_RADIOBUTTON
  - CMS_INPUT_TOGGLE
  - CMS_INPUT_TEXT
  - FS_BUTTON

  A relevant form in which a user role can be selected might, for instance, look as follows (see also Figure 6-15):

```
<CMS_INPUT_COMBOBOX name="role">
    <ENTRIES>
      <ENTRY value="privat">
        <LANGINFOS>
          <LANGINFO lang="*" label="Private customer"/>
          <LANGINFO lang="DE" label="Privatkunde"/>
        </LANGINFOS>
      </ENTRY>
      <ENTRY value="partner">
        <LANGINFOS>
          <LANGINFO lang="*" label="Partner"/>
        </LANGINFOS>
      </ENTRY>
      <ENTRY value="business">
        <LANGINFOS>
          <LANGINFO lang="*" label="Business customer"/>
          <LANGINFO lang="DE" label="Geschäftskunde"/>
        </LANGINFOS>
      </ENTRY>
    </ENTRIES>
    <LANGINFOS>
      <LANGINFO lang="*" label="Benutzerrollen"/>
    </LANGINFOS>
  </CMS_INPUT_COMBOBOX>
```

- **Evaluation in the FirstSpirit template**
  Evaluating within the preview calculation is not possible because this is carried out for a whole project and not for one user. The variables can be evaluated using JSP code or JavaScript.

- **Evaluation during runtime (JSP)**
  In the front-end server the input values can be output, for instance, using JSP code:

```
<%= session.getAttribute("fs.preview.role").toString() %>
```

  In this example, `role` is the variable name of the input component the editor uses to select the role of the website user (see example above).

- **Evaluation during runtime (JavaScript)**
  The input values can be output via:

```
WE_API.Preview.getParameter("role");
```

  See FirstSpirit Developer API, `de.espirit.firstspirit.webedit.client.api` package, `Preview` interface for more information. This package provides methods for settings parameter too.

7.5 API enhancements

The FirstSpirit API documentation describes the FirstSpirit interfaces which are used in the templates and scripts to access a huge variety of values, functions, etc.

For more information about changes concerning the development of custom modules please see Chapter 9.1 page 127.

7.5.1 FirstSpirit Access API

Some methods which had been set to "deprecated" in previous FirstSpirit versions have been omitted from FirstSpirit version 5.1. Overall in FirstSpirit 5.1, these API changes were kept to a minimum.

Methods can also be set to "deprecated" in version 5.1 itself. The methods concerned and the methods that should replace them in each case can be found in the FirstSpirit Access API. Methods with this state can still be used, but as they will be omitted in subsequent versions, this is not advisable. See also Chapter 10.2 page 137.

7.5.2 FirstSpirit Developer API

The FirstSpirit Developer API is stable in a minor version series, i.e. the methods available in version 5.1 may change with the next change in minor version (to 5.2).

The following API enhancements, among others, have been made:

The Developer API has been expanded in particular to include new interfaces and useful methods for **implementing project-specific reports** (packages `de.espirit.firstspirit.client.plugin.report` and `de.espirit.firstspirit.client.plugin`):

- The interface `ReportPlugin<T>` (`de.espirit.firstspirit.client.plugin` package) was updated in version 5.1. It can be used to define the basic properties of a report. The `isVisible()` method, for instance, can be used to control under which conditions the report is to be visible (e.g. only in SiteArchitect or only in ContentCreator). The `getDefaultItem()` method defines the action that occurs when the user clicks on a report entry.
- Thanks to the new `TransferHandler<T>` interface (`de.espirit.firstspirit.client.plugin.report` package), reports can now be edited using a variety of file types, which in turn can be used for drag-and-drop actions.
- The `DataRenderer<T>` interface (`de.espirit.firstspirit.client.plugin.report` package) provides, among other things, a detail view for report entries.

While the interface `WebeditReportPlugin<T>` (Package: `de.espirit.firstspirit.webedit.plugin`) has been used so far for the implementation of reports in ContentCreator, the implementation should be adapted to the interface `ReportPlugin<T>` (Package: `de.espirit.firstspirit.client.plugin`) in FirstSpirit Version 5.1. Background: With `ReportPlugin<T>` an interface has been created which can be used for ContentCreator as well as for SiteArchitect. In this context the interface `WebeditReportPlugin<T>` has been deprecated. See also documentation about the *FirstSpirit AppCenter*.

FirstSpirit Version 5.1 features enhanced additional **UI expansion options** in the FirstSpirit Developer API. In addition to already existing options (such as display of project-specific tool bar buttons and context menu entries), it is now also possible to use the `UIAgent` interface to access an element's object type icons that are of the

`IDProvider` type (via the `ImageIcon UIAgent.getIcon(IDProvider)` method). The returned `ImageIcon` type object can be used to display icons within plug-ins or reports in FirstSpirit SiteArchitect. Examples of using the functionality can be found in the *FirstSpirit online documentation* under "Plug-in Development"/"Implementation and Deployment"/"Using FirstSpirit APIs"/"Working With Store Elements"

In addition, the Content Creator JavaScript interface (`de.espirit.firstspirit.webedit.client.api` package) has been expanded—particularly the `Preview` interface inside. The purpose is to provide improved support for **dynamic HTML design in ContentCreator** (refer to section 7.4.5, page 103 for more information). The `repaint` method can be used to reposition the frame around editable areas if, for instance, the size or position of the affected HTML element was changed when the user clicked on it. The `rescan` method can be used to detect HTML areas that are reloaded by scrolling or clicking on them, for instance, and to add editing functions to them. The `editorId(...)` function in the relevant template is used to do this. The reload function, which has been available since FirstSpirit version 5.1, can also be used to reload only the area with the referenced `editorId` instead of the entire page as before. The `addElementReloadListener()` function can be used to listen to ContentCreator-driven changes to the HTML. If the ContentCreator reloads sections this way due to changes, the listeners registered for this will also be notified. For more information about the `Preview` interface, also refer to the FirstSpirit online documentation under "Plug-in development"/"Development and Allocation"/"Use of FirstSpirit APIs"/"JavaScript API (ContentCreator)"/"Preview".

Access to **objects of an external FS_LIST** (FS_LIST, Service type) was already possible using pre-existing methods. With the new `ExternalFormsProducer` interface (`de.espirit.firstspirit.access.store.templatestore.gom.fslist` package), the list can now also be filled with objects from the external source via the API.

The new `ModuleAdminAgent` interface (`de.espirit.firstspirit.agency` package) provides for API-supported **administration of modules**. Modules can therefore now be installed and updated easily using the API.

In addition, the **internal API** has been restructured in parts. As in the case of every new release, usages of API which is not released should be checked and ideally adapted to released API.

8 New/changed functions for administrators

8.1 IPv6 support

Version 5.1 brings IPv6 connectivity to FirstSpirit, making it ready for use with forward-looking network technologies. The new Internet protocol will affect features such as the server configuration, license validation, and even external software components such as the integrated Eclipse Jetty web server.

FirstSpirit supports simultaneous communication via IPv4 and IPv6 ("dual stack"). -D parameters can be used to configure whether the client/server should run primarily under IPv4 or IPv6 or only under one of the two versions (see section 8.1.1 page 110 (server) and section 8.1.2 page 110 (client) for more information). The parameters for the server are entered in the file `fs-wrapper.conf` as necessary; those for the client are entered in the connection settings (see *FirstSpirit documentation for administrators*, section "Configuring connection settings" and "Start page").

Using the configuration

```
-Djava.net.preferIPv4Stack=true
```

IPv6 use can be completely deactivated, i.e. all communication takes place via IPv4 for both the server and the client. This is the default configuration.

The current IPv configuration for the client or server is recorded in the log file `fs-server.log` at the start, e.g.

```
INFO  08.03.2013 09:09:35.381 (de.espirit.firstspirit.io.InetAddressUtil):
IPv4 preferred
```

Additional modes:

- IPv4 disabled
- IPv6 disabled
- IPv6 preferred

IPv6 is only supported in Microsoft Windows as of JDK version 1.6.0_34.

8.1.1 Server configuration

If the configuration file `fs-server.conf` specifies the parameter `SOCKET_HOST`, the server connects with all addresses found by the DNS query (IPv4 and/or IPv6).

With the configuration

```
Dfs.disableIPv4=true
```

the server only connects with the IPv6 address, even if there is an IPv4 DNS entry. The parameter is only evaluated if the parameter `SOCKET_HOST` is specified in the file `fs-server.conf`. This parameter is deactivated by default (`false`).

If the configuration file `fs-server.conf` does **not** specify the parameter `SOCKET_HOST`, the server listens to all addresses, regardless of whether they are IPv4 or IPv6. Exception: IPv6 is deactivated by the parameter `-Djava.net.preferIPv4Stack=true`.

8.1.2 Client configuration

The following -D parameters can be used for client configuration:

`-Djava.net.preferIPv4Stack=true`	The client only connects via IPv4.
`-Dfs.disableIPv4=true`	The client only connects via IPv6.
`-Dfs.preferIPv6=true`	If the server can be reached via IPv4 and IPv6, the client tries to connect via IPv6 first. By default, the client will try to connect via IPv4 first.

These parameters are deactivated by default (`false`).

If the server can be reached via IPv4 and IPv6 but only one connection is working, this connection is used automatically regardless of the setting configured via the -D parameters.

In order to specify a numerical IPv6 address in the connection settings of the client (e.g. via the FirstSpirit start page or the WebStart configuration in ServerManager), the address must be written in square brackets, e.g.

```
http://[fe80::222:4dff:fe7a:e6e6]:8080
-Dhost=[fe80::222:4dff:fe7a:e6e6]
-Dport=4088
```

8.1.3 Configuring the internal Jetty web server

The integrated Jetty web server also works in dual-stack mode, with IPv4 or IPv6. For this to work, the IP address must be configured as the host in the configuration file `fs-webapp.xml` and the desired listeners must be entered.

No host name is specified in the default configuration. Jetty then listens to both IP versions simultaneously. If a host name is specified, Jetty only connects its listener to the first address supplied for the host name and not to IPv4 and IPv6 simultaneously.

Example configuration:

```
<Call name="addConnector">
  <Arg>
   <New class="org.eclipse.jetty.server.nio.SelectChannelConnector">
    <Set name="port"><SystemProperty name="HTTP_PORT" /></Set>
    <Set name="host">2001:0:53aa:64c:38ca:211e:2be2:d633</Set>
    <Set name="maxIdleTime">30000</Set>
    <Set name="Acceptors">1</Set>
    <Set name="statsOn">false</Set>
    <Set name="lowResourcesConnections">1000</Set>
    <Set name="lowResourcesMaxIdleTime">500</Set>
   </New>
  </Arg>
</Call>
<Call name="addConnector">
  <Arg>
   <New class="org.eclipse.jetty.server.nio.SelectChannelConnector">
    <Set name="port"><SystemProperty name="HTTP_PORT" /></Set>
    <Set name="host">192.168.100.200</Set>
    <Set name="maxIdleTime">30000</Set>
    <Set name="Acceptors">1</Set>
    <Set name="statsOn">false</Set>
    <Set name="lowResourcesConnections">1000</Set>
    <Set name="lowResourcesMaxIdleTime">500</Set>
   </New>
  </Arg>
```

```
</Call>
```

8.2 Multiple server administrators

Whereas previously only the administrator who is automatically created when a FirstSpirit Server is installed (user ID 1, login: Admin) held all permissions for servers and applications, this role can now also be allocated to other users.

It is easy to assign temporarily and withdraw subsequently without having to publicize the confidential admin password or change it again afterward. Furthermore, every time a user logs in as a server administrator, this event is logged together with details of the user who logged in.

The following types of administrator are now available in FirstSpirit:

- **Administrator:** The "administrator" is the user who was automatically created with ID 1 when a FirstSpirit Server was installed; there is only one administrator per server. This administrator always holds all permissions for FirstSpirit servers and applications. However, this user can be blocked from accessing specific projects ("ServerManager"/"Project"/"Properties"/"Options"/"Block administrator"). The administrator is always a server administrator too; this role cannot be taken away from this user ("super administrator").
- **Server administrator:** Server administrators always hold all of the permissions of the administrator. The role of server administrator can only be assigned to a FirstSpirit user by the administrator or by another server administrator. There can be multiple server administrators per server.
- **Project administrator:** Project administrators always hold all permissions in projects in which they have been added to the default "Administrators" group. If a project administrator also has server administrator permissions, he/she will hold all permissions on the server and in other projects too.

8.2.1 Internal FirstSpirit user as server administrator

When creating a new (internal) user, the role of server administrator can be assigned by activating the "Server administrator" checkbox in FirstSpirit ServerManager in the "User" menu/"Create user":

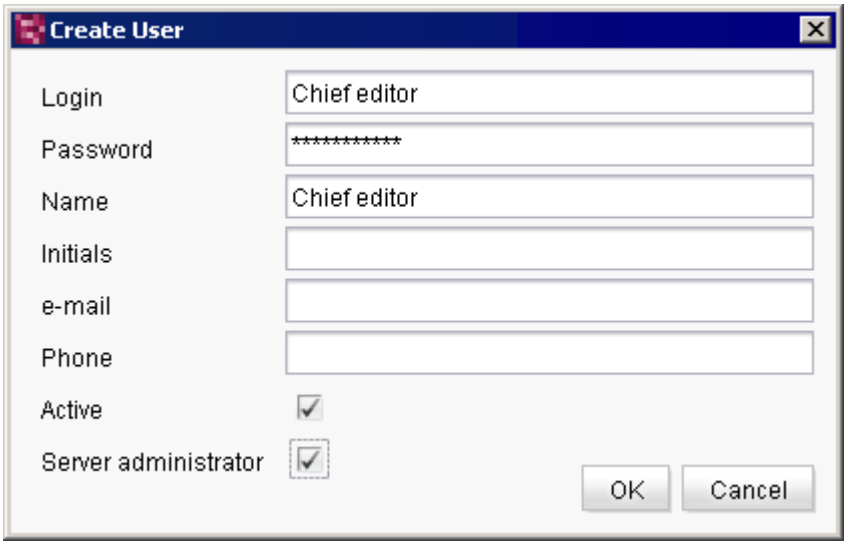


Figure 8-1: Creating users as server administrators

Existing (internal) users can be assigned the role of server administrator by activating the "Server administrator" checkbox in FirstSpirit ServerManager in the "User" menu/"Edit":

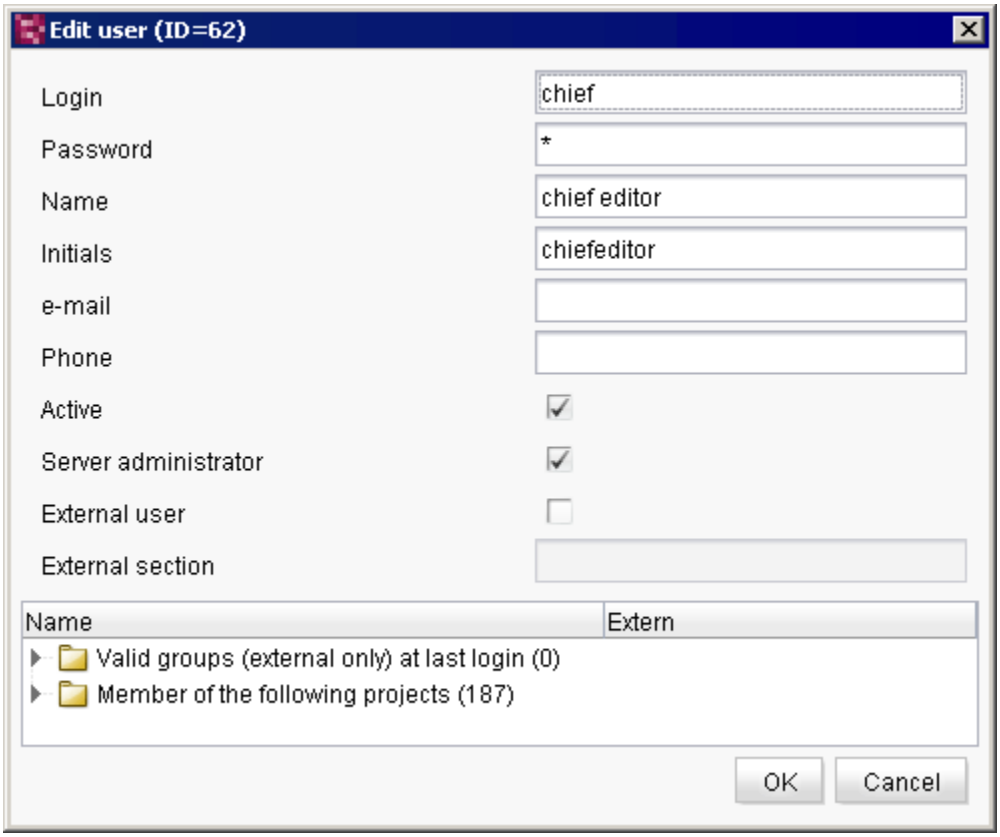


Figure 8-2: Assigning the "server administrator" role

This option can only be assigned by server administrators; initially, therefore, it can only be assigned by the administrator (user ID 1). If a user who had server administrator permissions on the other FirstSpirit Server is created by importing a project, this permission is removed during the import and must be regranted as necessary.

Exception: Administrator: The "server administrator" option is activated for the administrator (user ID 1) and cannot be deactivated.

8.2.2 External FirstSpirit users as server administrators

External users (for example deriving from LDAP) can be made server administrators via a corresponding parameter in the configuration file `fs-server.conf`:

```
externalServerAdminGroup=
```

An external group name can be entered as a value here. All members of this group will then receive server administrator permissions in FirstSpirit.

In order to create more than one external server administrator group, a unique extension is attached to the "externalServerAdminGroup" key, e.g.

```
externalServerAdminGroup.1=
externalServerAdminGroup.2=
```

Example for an LDAP definition of two external server administrator groups:

```
externalServerAdminGroup.1=CN=fs-crew,OU=FIRSTspirit,OU=Projekte,DC=e-
spirit,DC=de

externalServerAdminGroup.2=CN=fs-dev,OU=FIRSTspirit,OU=Projekte,DC=e-
spirit,DC=de
```

This configuration overwrites configurations that may have been set in ServerManager for the relevant users (see section 8.2.1 page 112).

The "server administrator" property is set for external users and group members every time they log in.

8.2.3 Who is a server administrator?

A list of the users who hold server administrator permissions is shown in FirstSpirit ServerManager under "User"/"Edit":

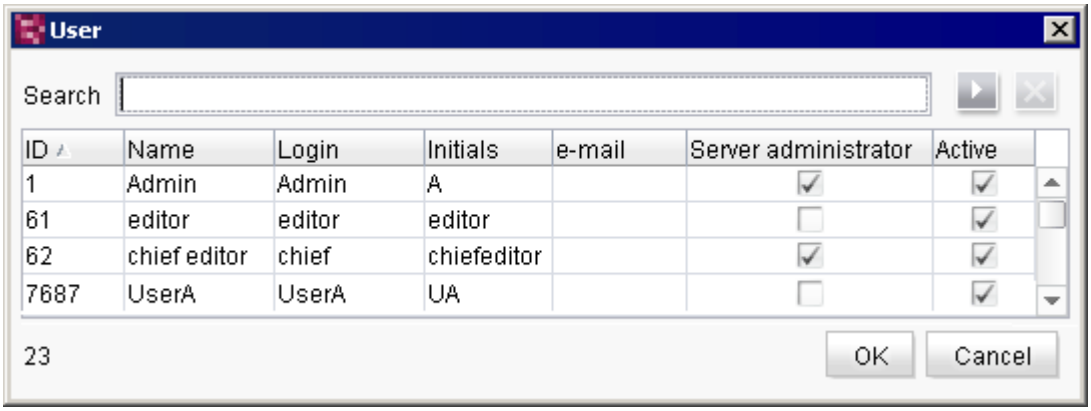


Figure 8-3: FirstSpirit users

The list can be sorted accordingly by clicking on the header of the "Server administrator" column. Please note that, for **external** users with server administrator permissions, this

list only reflects the state of the last FirstSpirit login and not the current LDAP state. This means that

- there may be more server administrators than are marked in the list with a check in the "Server administrator" column, but they have not yet logged onto the FirstSpirit Server via LDAP, or have not done so since authorizations were changed
- a user whose server administrator permissions have been withdrawn in the LDAP and who has not logged onto the FirstSpirit Server via LDAP since will still be shown as a server administrator.

If the server administration permissions are assigned to a user via ServerManager, this is recorded in the file `fs-server.log`, stating the user name:

```
INFO  02.10.2013 10:43:05.767
(de.espirit.firstspirit.server.usermanagement.UserManagerImpl): Setting user
'chief' server admin permission to true
```

When a user logs onto the FirstSpirit Server with server administrator permissions, this is also logged accordingly, stating the user name, e.g.

```
INFO  02.10.2013 09:05:21.113
(de.espirit.firstspirit.server.sessionmanagement.SessionManagerImpl): new
session (ID=5030863150308873085, user=chief, userID=62, type=MAIN) created

INFO  02.10.2013 09:05:21.113
(de.espirit.firstspirit.server.sessionmanagement.SessionManagerImpl): Session
with ID=5030863150308873085 bound to ip 192.168.100.212

INFO  02.10.2013 09:05:21.113
(de.espirit.firstspirit.server.sessionmanagement.SessionManagerImpl): User
'chief' login with server admin permissions, session ID=5030863150308873085
```

8.2.4 (De)activation in the various FirstSpirit applications

When a user logs onto the server with server administrator permissions, he/she has all permissions

- in ServerManager
- in ServerMonitoring
- in their own connections, set up via API

by default.

- In SiteArchitect and
- ContentCreator

the server administrator initially only has the permissions that the user would have without the server administrator option.

If the server administrator permissions are to take effect in SiteArchitect, this can be activated via the new menu entry "Project"/"Administrator mode":

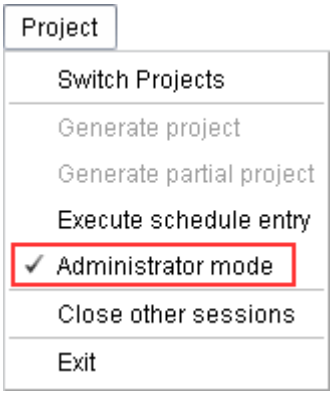


Figure 8-4: Activating the server administrator permissions in SiteArchitect

This menu entry is only available if the "server administrator" option is activated for the current user. This menu entry is deactivated by default. The activation only applies to the current project and the current session; when Site Architect is closed, the menu entry is deactivated again.

☑ Administrator mode If the current user is the administrator, the menu entry is activated and cannot be deactivated.

☐ Administrator mode If the "Block administrator" option is activated for the current project, the menu entry is available but cannot be activated.

If the menu entry is deactivated (i.e. the server administrator permissions are withdrawn) while some elements are still being edited, the following note is displayed: "Note: some objects are still in edit mode. Saving changes to these objects may only be possible with administrator privileges." In order to save the object(s), "administrator mode" must be reactivated as applicable; alternatively, editing mode can be closed with <Ctrl> + <Shift> + E, although the changes will not be saved.

If a similar function is required in ContentCreator, the `setAdminMode` method (**FirstSpirit Access API,** interface `User`, package: `de.espirit.firstspirit.access`) can be used to set "administrator mode" on a project-specific basis (although this method can only be executed by server

administrators). The `getAdminMode` method can be used to query whether "administrator mode" is set for the current user. The `isServerAdmin` method can also be used to check whether the current user is a server administrator.

If "administrator mode" is activated via API (`setAdminMode(true);`), this does not effect the "Administrator mode" menu item in the "Project" menu of SiteArchitect. The checkmark is not set as a result.

8.3 Configuring previews for ContentCreator

As Internet-enabled mobile devices such as notebooks, tablet PCs, and smartphones become more and more widespread, website designs need to be more and more flexible, with content which can be displayed perfectly on different display geometries and in different resolutions. Consequently, FirstSpirit now makes it easy for editors to check the display and navigation of website content with a variety of display sizes in the integrated preview. It also allows content, layouts, and images to be perfectly adapted to suit the output device concerned.

Along with size considerations, other aspects can also be taken into account, e.g. previews for specific user groups ("multi perspective preview").

8.3.1 Preview for different display sizes

In order to simulate different display sizes, a range of previews can be configured for ContentCreator for the current project by clicking the icon next to "Configure preview viewports" in the "Options" area of the project properties. The following window opens:

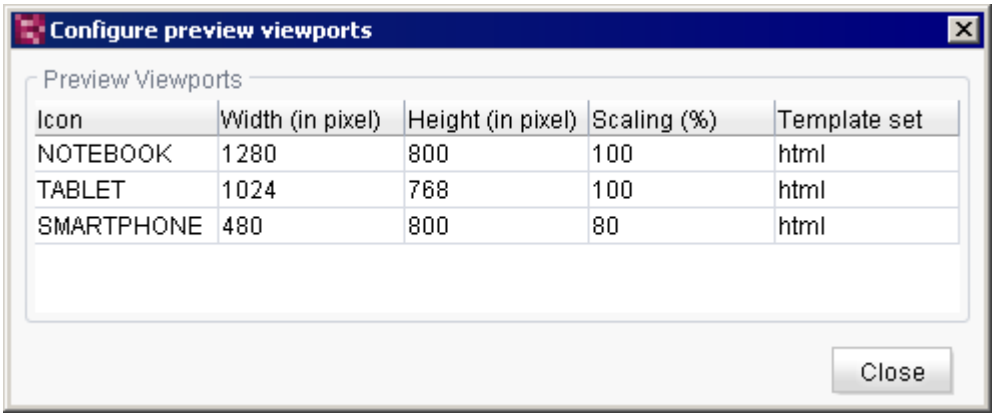


Figure 8-5: Project properties – Configuring previews

The three viewports "NOTEBOOK", "TABLET" and "SMARTPHONE" with the values shown in Figure 8-5 are pre-configured by default. Moreover, the viewport "DESKTOP" can be configured as well as further viewports with other values.

Icon: Specified identifier for the view. These identifiers are assigned icons which are displayed in ContentCreator and which allow the editor to identify the desired display size for checking the website content:

- DESKTOP:
- NOTEBOOK:
- TABLET:
- SMARTPHONE:

Width (in pixel): Width of the view in pixels. The default setting for the "DESKTOP" view is 1280 pixels.

Height (in pixel): Height of the view in pixels. The default setting for the "DESKTOP" view is 720 pixels.

Scaling (%): The preview can be scaled in the four available views in ContentCreator. The value specified here defines the scaling with which the relevant view is to be displayed initially. However, the editor is able to change the scaling. The default scaling for the "DESKTOP" view is 100%. For smartphones, on the other hand, 80% scaling provides a more realistic display.

Template set: From this drop-down list, the user can select a template set that is available for the project and is taken into account for the relevant view. By default, the template set which is selected for the ContentCreator (in the ServerManager / project settings, area "ContentCreator settings" / "ContentCreator template set"), otherwise the first in the area "Template sets".

Right-clicking with the mouse on the overview window opens the context menu:

The "New" context menu entry is used to add one of the four available views to the project. The window that opens displays the "DESKTOP" view with the default values (which are described above):

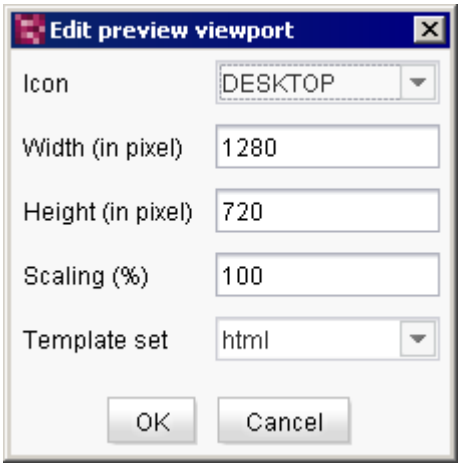


Figure 8-6: Project properties – Selecting a preview

Icon: Other views can be selected from this drop-down list. The values can be changed in the following fields depending on the project specification.

The following views are available:

- DESKTOP
- NOTEBOOK
- TABLET
- SMARTPHONE

The icons for the views are displayed in ContentCreator in the order in which they were created. In the case of the configuration shown in Figure 8-5, the icon for the "Desktop" view would therefore be shown on the left with the icon for the "Smartphone" view to the right of it.

The "Delete" context menu entry is used to remove the relevant view from the overview.

To edit a view, the user can click on the "Edit" context menu entry or double-click on the view. The window in Figure 8-6 opens.

8.3.2 Other preview perspectives

In addition to the display sizes for the various output devices, other aspects can also be simulated as a preview, e.g. user-specific or role-specific perspectives. The user can simply click to display the page content as it is seen by specific user groups, e.g. private customers, partners, or business customers.

The configuration is carried out using a page template in the relevant project. This must be selected in the "Preview parameters" field in the "Options" area of the project properties:

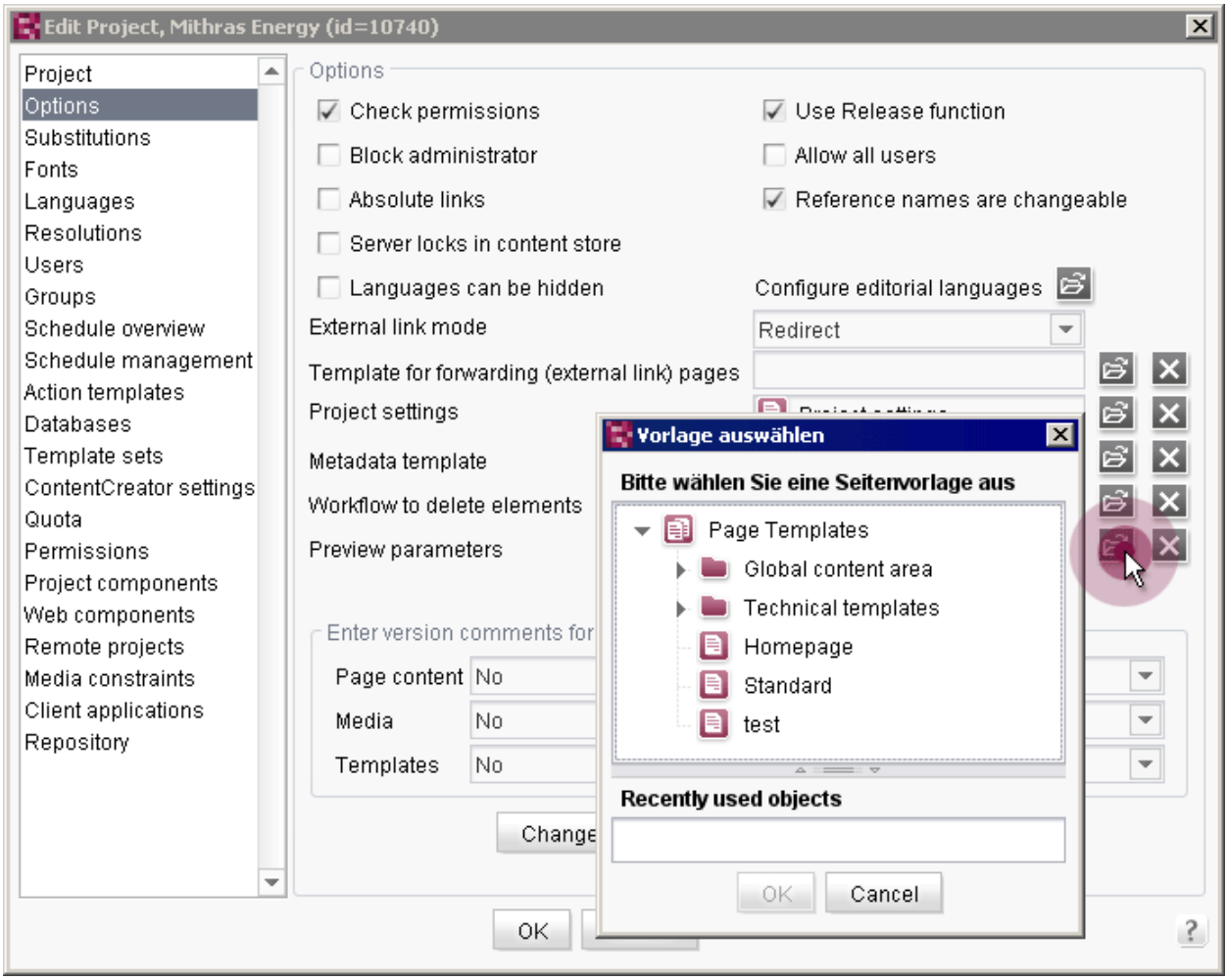


Figure 8-7: Selecting a page template for preview configuration

The input components defined in the page template are displayed in ContentCreator and can be filled in by the editor. The values entered by the editor can be used for the output of the current preview page.

For additional information on configuring the page template, see section 7.4.5, page 103.

8.4 Support for Berkeley DB 5

In addition to continuing to offer the reliable 3.x version of Oracle Berkeley DB as standard as a repository for saving content data, FirstSpirit version 5.1 will also include version 5.x in order to provide the latest performance advantages over the previous version in terms of technical development and to stay on top of future development trends. For reasons of compatibility, both versions can be used in tandem on the same server within different projects. Migration from one version to the other is as easy as selecting it from a list.

In order to operate different types of repository on a FirstSpirit Server (in different projects) at the same time, the repositories have been moved into modules:

- `fs-berkeleydb3.fsm`
- `fs-berkeleydb5.fsm`

This means that there is now an infrastructure available which can be used for other customer-specific repositories if required.

The required repository can be selected via the project properties (ServerManager) in the "Repository" area:

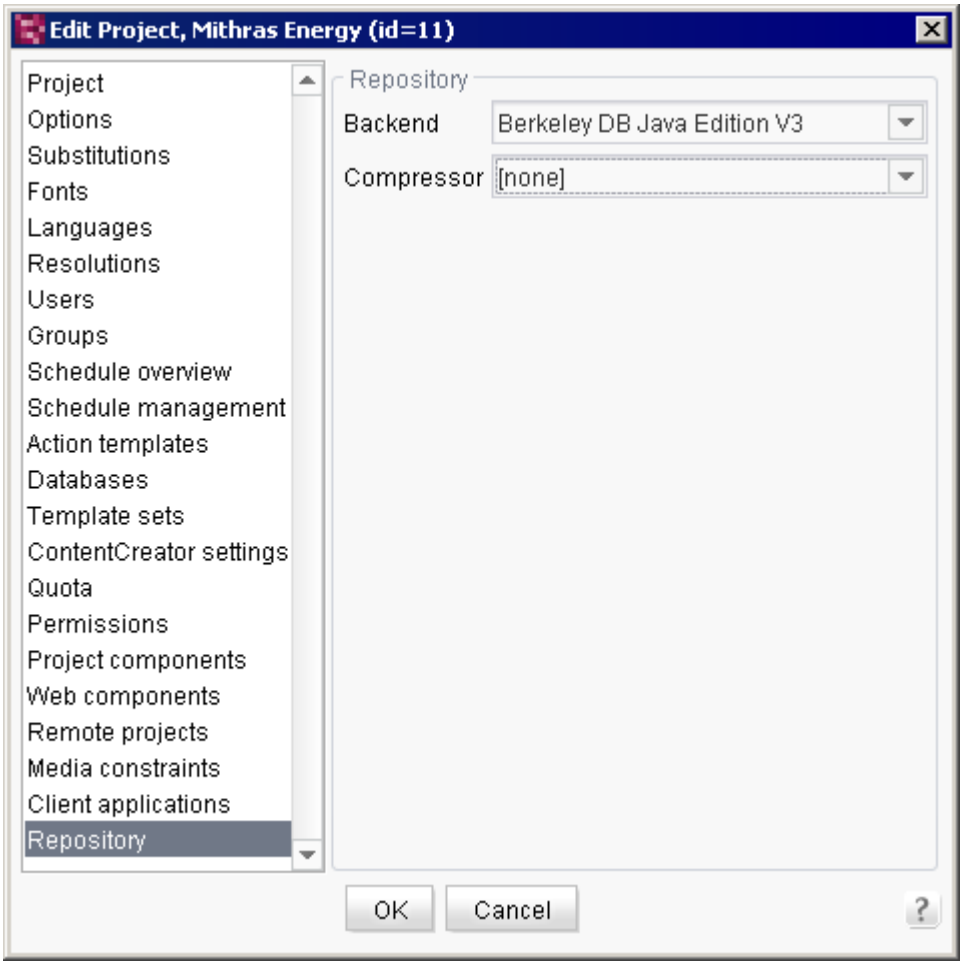


Figure 8-8: Project properties – Repository

Backend: The required repository can be selected from this drop-down list.

Compressor: A different compression can be set as necessary via this drop-down list. Compression affects both the amount of disk space required and the access speed.

- [none]: No compression is used (recommended default setting).
- Deflate: Algorithm with high compression ratio
- Snappy: Algorithm developed by Google and designed to provide high speeds
- LZ4: Algorithm designed to provide high speeds

When the user confirms the selection with "OK", the system starts to convert the data using the desired settings. The relevant project is deactivated during the process.

To prevent any data loss, anyone using the project should log off first.

The new Berkeley version 5.x was officially released following extensive quality assurance checks; however, as this is a significant version upgrade and it concerns a third-party product, this version should be tried out on a test system before being launched on production systems.

8.5 Report plug-ins

The checkboxes used in FirstSpirit version 5.0 to activate report plug-ins have been removed in FirstSpirit version 5.1. In line with the general module implementation, a report plug-in is activated in FirstSpirit version 5.1 by installing the menu on the FirstSpirit Server:

- In FirstSpirit SiteArchitect, the classes included in the module are automatically reloaded once the module has been installed on the server and they are then available in the project development environment without any further configuration. If this is not desired, the report developer must disable this function in the module implementation (see *FirstSpirit online documentation*, plug-in development for more information).
- Dynamic class loading is not possible in FirstSpirit ContentCreator. The module must first be added as a web component (in the "ContentCreator" tab) and installed on the relevant web server (see *FirstSpirit documentation for administrators*, section "Web components"). The report classes will then be available in the editing environment.

Note on compatibility: Modules that were developed for FirstSpirit version 5.0 should be compatible with version 5.1 in principle. As the two-stage activation of report plug-ins in version 5.0 ([1] installation and [2] activation via the checkboxes that have now been omitted) has changed with version 5.1, all modules installed in FirstSpirit version 5.1 are now immediately active (see above).

8.6 Other changes

The **order of resolutions** in the project properties (ServerManager/Project/"Resolutions") can be changed using the context menu, the mouse pointer (drag-and-drop), and the keyboard. The arrow keys can be used to navigate within the table and the rows can be moved by holding down the <Ctrl> key.

The **Google Web Toolkit** (GWT) used for ContentCreator has been updated from version 2.4 to 2.5.

The license for the **Java wrapper** has been updated so that all wrapper versions that come out before 06/11/2014 can now be used.

9 New/changed functions in modules

Version 5.1 provides API-supported administration of modules using the new `ModuleAdminAgent` interface (`de.espirit.firstspirit.agency` package; see section 7.5.2, page 107).

9.1 Develoment of custom modules

The definition of a `<wep-app>` definition in the file `module.xml` is necessary for the successful communication between a custom input component and a service. This was not necessary up to FirstSpirit Version 5.1 if the respective module was installed on the integrated Jetty web server or in the "Preview" web component. From Version 5.1 on, the `<wep-app>` definition is mandatory also in these cases.

Otherwise this could lead to the following exemplary error message:

```
de.espirit.firstspirit.access.ServiceNotFoundException: Service
'<custom service interface name>' not found

...

Caused by: java.lang.IllegalStateException:
java.lang.ClassNotFoundException: <custom service interface name>

...

Caused by: java.lang.ClassNotFoundException: <custom service
interface name>
     <potentially long part from web container code>
     at java.lang.ClassLoader.loadClass(ClassLoader.java:358)

...
```

`<wep-app>` definition:

```
<module>
          ...
          <components>
                    ...
                    <web-app>
                              <name>...</name>
                              <displayname>...</displayname>
                              <description>...</description>
                              <web-xml>web.xml</web-xml>
                              <web-resources>

<resource>lib/your.jar</resource>
                              </web-resources>
                    </web-app>
                        ...
          </components>
        ...
</module>
```

Relevant are the `web-resources` line which must reference your JAR file and the `web-xml` line.

The `web.xml` file itself may have minimal content:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/j2ee"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app 2 4.xsd" version="2.4">
    <display-name>...</display-name>
    <description>...</description>
</web-app>
```

The structure of your `FSM` file should then (at least) look like this:

```
+ lib
    - your.jar
+ META-INF
   -  MANIFEST.MF
   -  module.xml
- web.xml
```

Next step is to add and deploy the web application (at least into the "Preview" scope). See *FirstSpirit Documentation for Administrators*.

9.2 FirstSpirit Content Transport

FirstSpirit data can be bundled into a package using the "FirstSpirit Content Transport" module function. This allows the data to be imported into other projects and onto other servers.

FirstSpirit version 5.1 allows the user to save to different locations which can be used to store created ZIP files. In addition to the **local file system** and **network drives**, **external storage locations** can be used as well, including Internet-based locations such as Dropbox. To be able to use external storage media, the relevant modules need to be created. This type of module would then handle Internet service authentication, for instance.

The storage locations can be configured in the project properties in the ServerManager under "Project components"/"FirstSpirit Content Transport Storage App" (refer to "Configuring storage locations" in the *FirstSpirit CorporateContent* module documentation for more information).

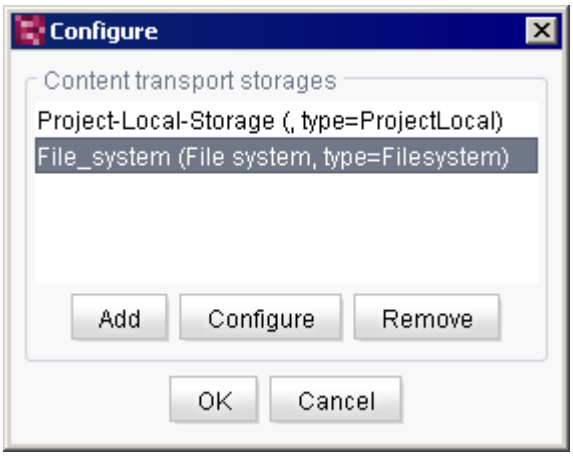


Figure 9-1: Configuring storage locations for Content Transport content

The "FirstSpirit Content Transport Storage App" project component is installed automatically during a new install or a server update to version 5.1 if the `license.PACKAGEPOOL` license key is present in the `fs-license.conf` license file with the value `1`.

The storage locations configured using this project component are then offered as selection options when generating and installing feature ZIP files in SiteArchitect (refer to "Content Transport" in the *FirstSpirit CorporateContent* module documentation for

more information):

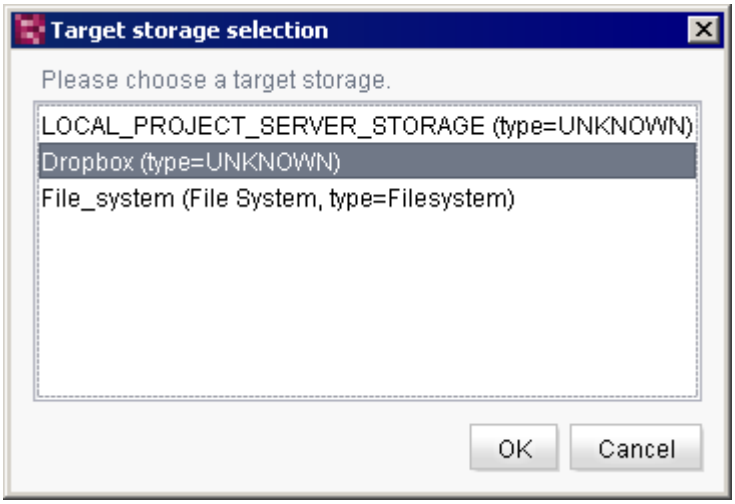


Figure 9-2: Choosing the storage location

FirstSpirit 5.1 also offers the ability to **automatically update** (generate and install) Content Transport content at predefined points in time using FirstSpirit schedule entry planning. For this purpose, the new "Content Transport (create, update, install)" action in "Schedule management" in the ServerManager project properties is used:

- to regularly provide export of an existing feature (combine FirstSpirit content) with a current state and installation of other projects ("push"):

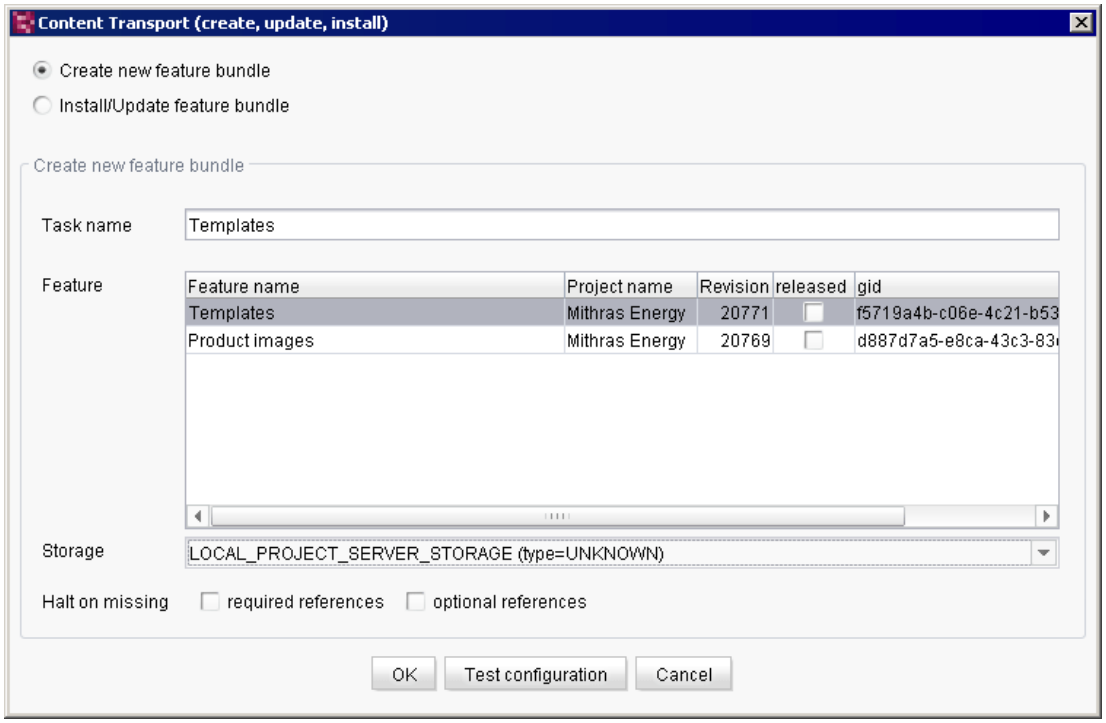


Figure 9-3: Content Transport schedule – Create

- to regularly import existing Content Transport content from one storage location to a project ("pull"):

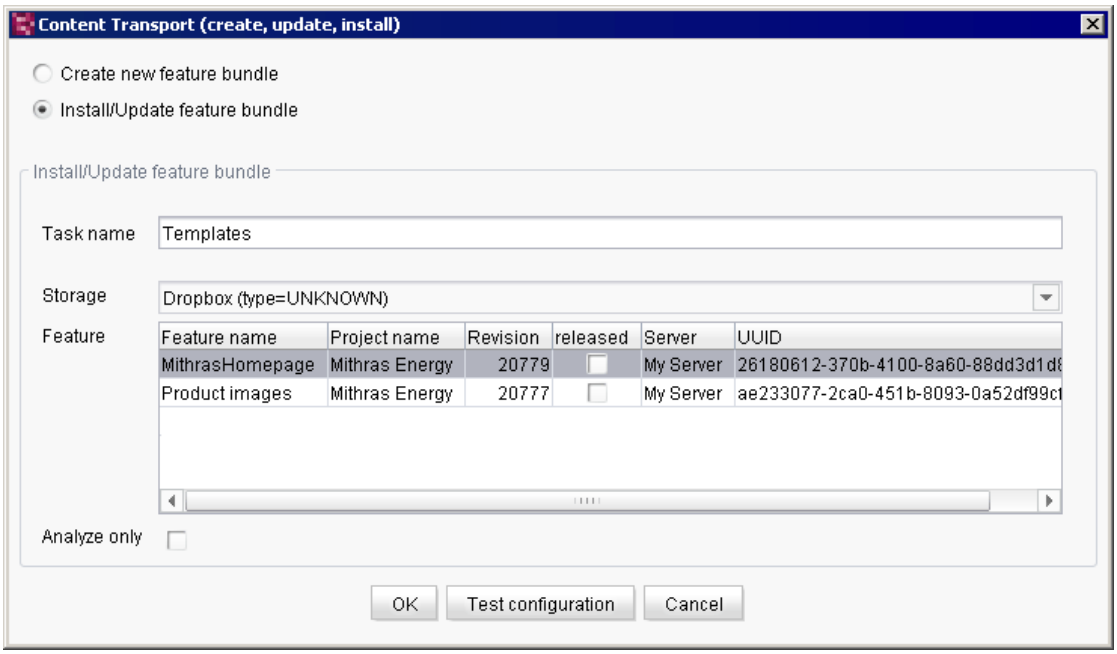


Figure 9-4: Content Transport schedule – Install/Update

Refer to "Automation" in the *FirstSpirit CorporateContent* module documentation for more information.

In addition, FirstSpirit objects can be added easily to the "Included objects" field of a feature using **drag-and-drop**.

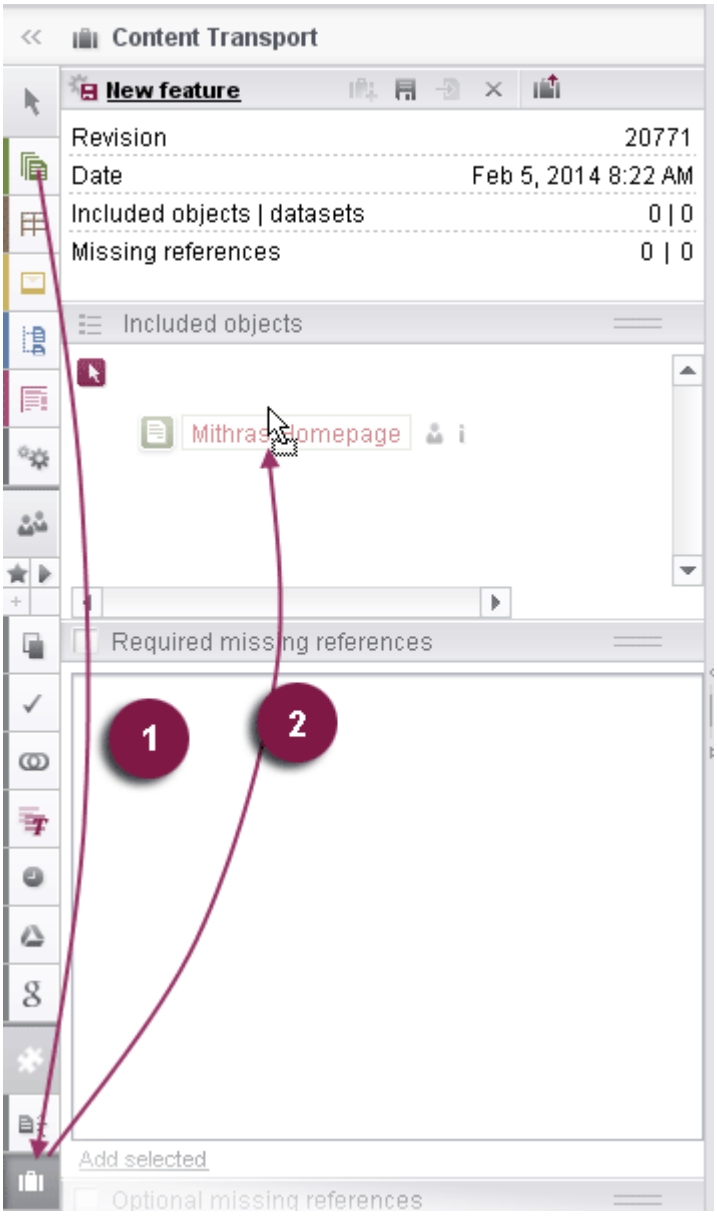


Figure 9-5: Integrating a page in a feature using drag-and-drop

10 Appendix

10.1 Changes in software behavior

- **Preview of media:** Whereas media were previously always given the file name

```
x.[fileextension]
```

  in the preview, the reference name is used as of version 5.1, e.g.

```
construction_of_a_solar_system.jpg
```

- **"Search" menu:** Certain search options have been omitted from the main "Search" menu in SiteArchitect. They have been replaced with the global search function that has been implemented/further developed in FirstSpirit V5.0. In addition, the new filter options can also be used, e.g. filtering by template type via the "Path" or "Type" facet, by metadata via the "Metadata" facet, etc.
  - *Unreleased objects:* The system filter "Search not released elements" below "References/Elements" (under "Filters and Sources") can be used to search for objects that have not been released. The results can then be filtered by store via the "Path" facet under "Limit search results (Facets)".
  - *Search in templates:* Unlike in the "Search for templates" function included up to V5.0, it is possible to search for terms linked with AND in the global search. The "Search in templates" filter (under "Filters and Sources") is used to access the relevant tab for the corresponding element directly from the search result.
- **Metadata in reference graph:** In FirstSpirit 5.1, the reference graph (e.g. SiteArchitect, context menu "Extras"/"Display dependencies" or keyboard shortcut <Ctrl> + R) now also shows references to objects which result from the metadata. This means that more referenced objects may now be displayed/returned than in previous FirstSpirit versions.

- **"Extras" menu:** The "Print" function has been omitted from the main "Extras" menu in SiteArchitect. The  icon can be used to print workflows ("State diagram" tab) and database schemata.
- **"View" menu:** The "Display area of the AppCenter" entry underneath the main "View" menu in SiteArchitect is now called "Display area of AppCenter / Preview".
- **Browser Engine:** With FirstSpirit Version 5.1 version15 of Mozilla Firefox for use within the integrated preview is now supported officially. Version 3.6 is deprecated and no more supported. But it can still be used for a transitional period. A special version of Firefox integrated in SiteArchitect is used for use of Mozilla Firefox so that it is not necessary to install it locally on the workstation.
- **Language tabs of pages and sections:** The check marks on the language tabs of pages and sections in the Page Store can be used to set the status "Page is completely translated to this language" and "include this section in the output":

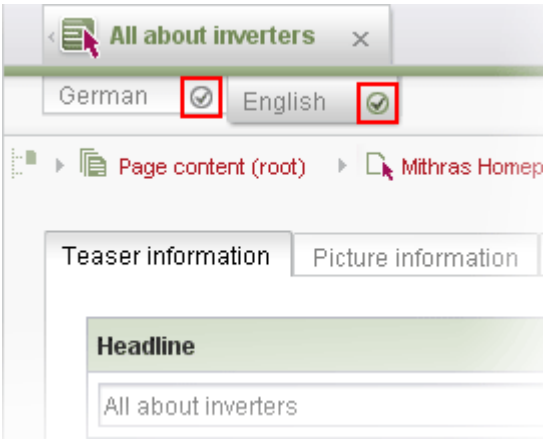
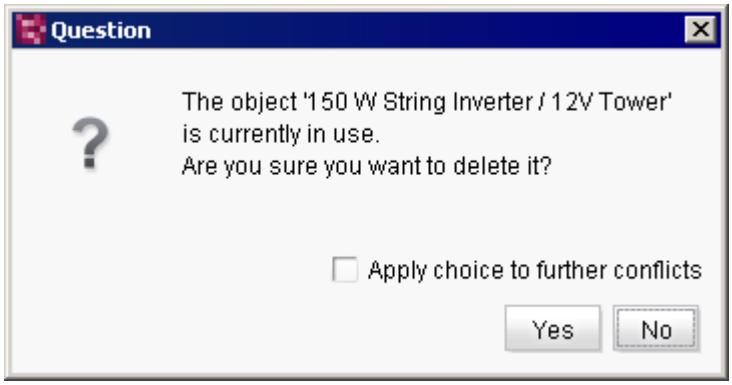


Figure 10-1: Settings on section level

In order to change the status for a language tab which is in the background at the moment (the tab "German" in Figure 10-1) the language tab must first be brought into the foreground by clicking on it. The check mark can then be removed or set again with another click.

- Display of **filtered data sources** in the tree structure: Previously, in addition to the display name of the data source, the name of the database query was displayed (in parentheses) for permanently filtered data sources (context menu entry "Extras/"Set filter" to data sources). This is no longer the case in version 5.1. The filtered data source should therefore be given a meaningful display name. The associated icon has also been updated:  .
- **Deleting of datasets in use** (only for server administrators): If more than one datasets, from which at least one is still referenced, are selected at the same time and then deleted by a server administrator, the following dialog is now displayed

after having confirmed the question "Do you really want to delete the selected data records ?" with "Yes":



  - If the option "Apply choice to further conflicts" is **activated**, all selected datasets will be deleted without any further request with clicking on the "Yes" button. If you click the "No" button, only those datasets are not deleted which are still used. They will be listed in the following dialog.
  - If the option "Apply choice to further conflicts" is kept **deactive**, the current dataset will be deleted if you click the "Yes" button. If you click the "No" button, it will not be deleted. This question will then be displayed for all datasets which are selected and still in use.

- **Translation help:** If modifications were made to content within the translation help (menu "Extras" / "Translation help"), there will be shown a prompt if you would like to save or discard the changes if you close the translation help by a click on the x icon.
- **FS_LIST, type DATABASE:** As of version 5.1, the  icon is used for the "Add" action in input components of the type FS_LIST, type database, rather than  as previously. If the column width is specified using the *width* parameter (within <*COLUMNS*> / <*COLUMN*>), the column width can still be changed by the editor temporarily (as long as "resize="no" is set). The column width set manually will now also remain in View and Editing modes until it is changed again by the editor.
- **Selecting the next editor in ContentCreator:** If previously the name or part of the name of the desired editor was required in the workflow dialog when selecting the editor, a list of available users can now also be opened by clicking in the "Next editor" search field.
- **Spell check in ContentCreator:** If the spell check is activated in ContentCreator ( icon), new words and character strings are also checked. Previously, the check had to be reactivated by clicking on the icon.
- **Security feature:** Access to the file `fs-client.jar` is now protected, which means that only registered FirstSpirit users may access it. A separate entry is created in the Java Web Start jar file cache for each user/login, which means that

  - if various different logins are used, the resources will be downloaded again for each login
  - if the server name or URL changes, Web Start will load all resources again.

- **License management:** FirstSpirit version 5.1 includes stricter checking of compliance with license restrictions. A license conflict (a license with `license.SCOPE=SINGLE` used by multiple servers) will now cause termination of the FirstSpirit server. Only the first server started with this license will not be terminated. All others will terminate after appropriate log notification with a 30 minute delay. To prevent the server from shutting down, a valid license needs to be imported to the affected server (refer to "License" in the *FirstSpirit documentation for administrators* for more information). Importing a different, valid license on another FirstSpirit server on the network will not suffice. If a license conflict arises when switching between two versions of FirstSpirit, the affected license can be used on the server again only after restarting the server.
- **Workflows:** The license parameter `license.WORKFLOW` controls whether only the standard workflows supplied with FirstSpirit ("Assign task" and "Request release") may be used on a particular FirstSpirit server (`license.WORKFLOW=0`) or whether users may also create and use their own workflows (`license.WORKFLOW=1`). The parameter `license.WORKFLOW=0` does not permit any workflows to be edited or created and only the workflows supplied as standard may be used. This parameter is now given greater consideration, which means that users may no longer start their own workflows or modified workflows, or switch these workflows to another state, following the update to version 5.1.
- **Project properties/line break:** Line breaks from the template sets are coded in FirstSpirit independently from the operating system as line feeds (0x0a). Selecting the operating system (previously in ServerManager under Project/Properties/Template sets) is therefore no longer required. The "Line break" combo box has been removed.
- **Report plug-ins:** The checkboxes used in FirstSpirit version 5.0 to activate report plug-ins have been removed in FirstSpirit version 5.1. In line with the general module implementation, a report plug-in is activated in FirstSpirit version 5.1 by installing the menu on the FirstSpirit Server. See also section 8.5, page 125 for more information.
- **Cache memory requirement:** The slider in the project properties (ServerManager/Project properties/Options), which was used in previous FirstSpirit versions to change the percentage of memory allocated for the Berkeley cache of the individual projects on a server, has been removed because all projects have shared a common Berkeley cache by default since FirstSpirit version 5.0.

- **URLCreator development:** Previously it was only possible to use lower case to specify the configuration parameters (e.g. `useiris`, `useregistry`, `usewelcomefilenames`) in a schedule script for a URLCreator implementation of a generation action. As of version 5.1, it is now possible to use upper case as well for improved clarity and other uses (e.g. `useIRIs`, `UseIris` or `UseIRIs`).

10.2 Discontinued functions in FirstSpirit version 5.1

- **API deprecations:** All functions of the FirstSpirit Access API, the discontinuation of which was announced in FirstSpirit Version 4.2 or before then ("deprecations", cf. API documentation) have been removed in FirstSpirit Version 5.1. For information about further API changes please see Chapter 7.5 page 106.

10.3 Notices for future versions

- Version 5.2 development will also focus on the FirstSpirit **developer experience** and will continue to improve on support for development processes. *For more information see FirstSpirit Roadmap 2013 - 2017*.
- Deprecation of the **getStoreElement()** method: Display logic can be implemented for workflows and scripts. Previously the `getStoreElement()` method of the `ClientScriptContext` interface (FirstSpirit Access-API, `de.espirit.firstspirit.access` package) was frequently used for this in the Content Store and for the respective scripts. This method was deprecated in FirstSpirit version 5.1 and is not present in subsequent versions. Instead, the newly implemented `getElement()` method with the same interface should be used. Unlike `getStoreElement()`, `getElement()` provides a dataset instance in which `getStoreElement()` returns a Content2 instance.