FirstSpirit™
Your Content Integration Platform

FirstSpirit™ Release Notes

FirstSpirit™ Version 4.2

Version	**1.05**
Status	**RELEASED**
Date	**2009-11-26**
Department	FS-Core
Author/ Authors	B.Ehle
Copyright	2009 e-Spirit AG
File name	RELN42EN_FirstSpirit_Releasenotes

e-Spirit AG

Barcelonaweg 14
44269 Dortmund | Germany

T +49 231 . 286 61-30
F +49 231 . 286 61-59

info@e-spirit.com
www.e-spirit.com

e-Spirit^AG

Table of contents

1 Introduction

This document introduces the new functions implemented in FirstSpirit™ Version 4.2. It is assumed that the reader is familiar with FirstSpirit™ and has background technical knowledge. To understand Chapter 5 and 6 in particular, in-depth knowledge of the respective field is required, for example template development.

1.1 Chapter overview

Chapter 2: Introduction to the new FirstSpirit functions, subdivided by user groups (from page 6).

Chapter 3: Overview of the system requirements for "FirstSpirit Version 4.2" (from page 13).

Chapter 4: This chapter describes new functions and modified operating sequences / user input sequences, which are relevant for editorial users (from page 16).

Chapter 5: Template development changes in FirstSpirit Version 4.2 concern, among other things, the link templates area and the configuration of table templates and content areas.(from page 76).

Chapter 6: New administrative functions are available via the server and project configuration application and via the new "FirstSpirit Enterprise Backup". This chapter also contains important information about the subject long-term archiving and backup with FirstSpirit (from page 101).

Chapter 7: Installation of a FirstSpirit server Version 4.2 and the procedure for updating a FirstSpirit Version 4.1 server are described in this chapter. In addition, the chapter contains important notes on updating and downgrading FirstSpirit Version 4.2 (from page 109).

Chapter 8: This chapter lists, in key point form, the expiring functions for the current Version 4.2 and the subsequent FirstSpirit Version 5.0 and points out planned changes (from page 109).

2 FirstSpirit 4.2 – An overview of the innovations

2.1 Overview

The new functions and enhancements in FirstSpirit Version 4.2 are intended to prepare existing projects for the new major release of FirstSpirit Version 5.0. The new concepts introduced can still be used in FirstSpirit Version 4.2 in parallel with the functions used to date (in Version 4.1). With the release of FirstSpirit Version 5.0, individual functions will then no longer be supported, but will instead be replaced by new, enhanced concepts. An example is the introduction of "generic link editors" in FirstSpirit Version 4.2. These provide greater functional scope and more flexible design options than the static editors used to date. The static editors will no longer be supported from FirstSpirit Version 5.0. The introduction of "generic link editors" in FirstSpirit Version 4.2 therefore also helps to migrate existing projects to FirstSpirit Version 5.0.

During the development of FirstSpirit Version 4.0, a lot of time was spent on the conception and implementation of the Migration Wizards to keep the time required to adjust a project to Version 5 conditions as short as possible. The Wizards supports the migration of existing projects; however, it cannot completely replace manual migration effort. Manual adjustment will also continue to be necessary in individual cases.

A further objective of Version 4.2 is consistent and logical further development of user prompting already begun with FirstSpirit Version 4.1. The focus is on new operating concepts and a clear, modern interface design. Apart from the redesign of the FirstSpirit JavaClient, the release of FirstSpirit Version 4.2 will also include a (design) relaunch of the FirstSpirit WebClient. To this end, the WebClient interface has been completely revised and replaced by new user prompting concepts ("Easy-Edit") and a more modern design (see Chapter 4.2 page 67). With the release of FirstSpirit Version 5.0, the revised design in WebEdit will also be followed by the use of new technologies.

In FirstSpirit Version 4.2, the FirstSpirit applications (JavaClient, server and project configuration) are switched over to the new "look & feel", introduced with FirstSpirit Version 4.1. With this, support for the classic "look & feel" is dropped.

FirstSpirit Version 4.2 will probably be the final release of the FirstSpirit-V4 line (before the launch of FirstSpirit Version 5.0).

2.2 For editors

Multi-Tabbing: The introduction of tab navigation in the editing area of FirstSpirit JavaClient enables convenient and clear editing within several workspaces. This function is available in JavaClient only (see Chapter 4.1.1 page 17).

Breadcrumb navigation: Above the form area, the path from the store root up to the current element is displayed in the form of "breadcrumb navigation", which can be used to perform actions using the context menu or for navigating to adjacent elements. This function is available in JavaClient only (see Chapter 4.1.2 page 23).

Display of the Stores as navigation areas: The complete project display used to date in the left-hand part of the FirstSpirit JavaClient's navigation is replaced by clearer individual display of the stores. This option is available in JavaClient only (see Chapter 4.1.3 page 24).

Integrated preview in JavaClient: The integrated preview provides the editor with a direct WYSIWYG preview of their page within FirstSpirit JavaClient. Apart from the form-based workspace in JavaClient, at the same time the content of the page is displayed in the integrated browser. This function is available in JavaClient only (see Chapter 4.1.4 page 25).

Content Highlighting: Based on the integrated preview, "Content Highlighting" provides further help for editors. If the editor is in the form-based workspace of the JavaClient, the currently edited form field is highlighted in colour in the integrated preview. If, on the other hand, the editor clicks an element in the preview, for example, a text passage, the corresponding section is automatically opened in a workspace and the editing option is highlighted in the form area. This function is available in JavaClient only (see Chapter 4.1.5 page 30).

Media gallery: Enhancement of the CMS_INPUT_CONTENTLIST input component enables several media to be selected from the project's Media Store and to be arranged and displayed within a gallery view This function is available in JavaClient only (see Chapter 4.1.6 page 35).

Generic link editors: Links are no longer maintained by means of static input dialogs, but instead using flexibly configurable, generic editors. This can result in differences in the display and user prompting of the input screen forms for the editors, which can be adjusted in a project specific way to a far greater extent than to date. This function is available in JavaClient and in WebClient (see Chapter 4.1.7 page 41).

Configurable, multi-line views within the Content Store: The display of content within the Content Store has also been completely revised. To this end, with FirstSpirit Version 4.2, options for configuring the views are made available within the Template Store, for example, for multi-line display of datasets or for showing pictures This function is available in JavaClient only (see Chapter 4.1.8 page 42).

Full text search in the Content Store: From FirstSpirit Version 4.2, full text search is available in the tables view of the Content Store, which searches through all text content in a table, including the texts deposited in input components of the type DOM Editor. This option is available in JavaClient only (see Chapter 4.1.9 page 44).

New Drag & Drop function for FirstSpirit objects: Several new "Drag & Drop" functions for FirstSpirit objects have been implemented within the scope of FirstSpirit Version 4.2. Among other things, media from the thumbnail view of a workspace can be copied into a picture input component of another workspace by means of Drag & Drop. This function is available in JavaClient only (see Chapter 4.1.10 page 45).

Restore settings on restart: The new "View" menu in FirstSpirit JavaClient can now be used not only to restore the window size and position but also to restore the positions of the separator between the tree and workspace and the workspace and preview on restarting the JavaClient. This function is available in JavaClient only (see Chapter 4.1.11 page 53).

Enhancements to the CMS_INPUT_OBJECTCHOOSER input component: Enhancements to the CMS_INPUT_OBJECTCHOOSER input component have been realised in FirstSpirit Version 4.2, for example, a new selection dialog for datasets and the option of switching directly to the referenced dataset in the Content Store. This function is available in JavaClient (see Chapter 4.1.12 page 54) and - with limited scope - in WebClient too (see Chapter 4.2.4 page 71).

Extension of reference graphs for format templates: Format templates used within the input components CMS_INPUT_DOM and CMS_INPUT_ DOMTABLE can now also be displayed using the reference graphs. This function is available in JavaClient only (see Chapter 4.1.13 page 56).

Status display in the JavaClient's workspace: A status display for the respective object is displayed in the active workspace of the JavaClient. This function is available in JavaClient only (see Chapter 4.1.14 page 57).

New quick text search in JavaClient: The search dialog for quick text search has been considerably enhanced. In addition, the dialog is no longer modal; this means that the editor can continue working in JavaClient without first closing the search dialog. This function is available in JavaClient only (see Chapter 4.1.15 page 58).

FirstSpirit project homepage: The familiar colour concept of the "Open Tasks" task list has now been implemented in the FirstSpirit project homepage too. This function is available in JavaClient only (see Chapter 4.1.16 page 60).

New Start dialog for the FirstSpirit JavaClient: The Start dialog of the FirstSpirit JavaClient has also been revised within the scope of the changeover to the new look

& feel. This function is available in JavaClient only (see Chapter 4.1.17 page 61).

New keyboard shortcuts: New keyboard shortcuts have been implemented within the input component **CMS_INPUT_DOM** for fast implementation of actions using the keyboard. This function is available in JavaClient and in WebClient (see Chapter 4.1.18 page 61).
The **version history** can now be opened by using the keyboard shortcut Ctrl + H. This function is available in JavaClient only (see Chapter 4.1.18 page 61).

Re-design of WebClient: The WebClient has been comprehensibly redesigned for WebEdit 4.2 with new buttons, adjustment of the layout within selection dialogs and a preview within picture input components. This function is available in WebClient only (see Chapter 4.2.1 page 67).

Easy-Edit: WebEdit 4.2 has been enhanced to include a new "Easy-Edit" function. The editor works directly in the project's preview in the browser where they can edit defined areas ("Hotspots") of the page. The "Easy-Edit" function has been introduced to enable direct editing of content within the preview page without using separate windows (pop-up windows). This function is available in WebClient only (see Chapter 4.2.2 page 68).

Support for inline tables in WebEdit: Since FirstSpirit Version 4.1, so-called "inline tables" can be integrated in the text flow by enhancement of the DOM Editor input component (CMS_INPUT_DOM) in JavaClient. The function is now available in WebEdit 4.2 – albeit with limited scope (see Chapter 4.2.5 page 72).

User support on connecting workflows: Connecting workflows is supported in FirstSpirit Version 4.2 by a new Help function (see Chapter 4.1.19 page 62).

Show properties of an object: Using the keyboard shortcut Alt + P or the context menu “Extras – Show properties“ technical and editorial information about particular project contents can be displayed in a separate dialog and summarised in a system report (see Chapter 4.1.20 page 63).

Central bug collection and system report: From FirstSpirit Version 4.2 a new infrastructure for collecting bugs and exceptions is provided. They can be summarised in a system report and then sent per mail (see Chapter 4.1.21 page 64).

Restrictions of the new functions to JavaClient or WebClient are described in Chapter 4.3 (page 74).

2.3 For template developers

Integrated form preview in the Template Store: Apart from the display of editorial content, the integrated preview can also be used in the Template Store. If a template is selected, the preview of the form area appears in this part of the form area with the defined retrieval values of the input components (see Chapter 5.1.1 page 76).

Generic link editors: From FirstSpirit Version 4.2, the configuration options for links are considerably enhanced by the introduction of generic link editors. The configuration of link templates can then, analogous to page and section templates, be created by adding input components into the form area (see Chapter 5.1.2 page 77).

Configurable, multi-line display of datasets: With FirstSpirit Version 4.2, the display of content within the Content Store has been optimised for editorial work. To this end, options for configuring the views are made available within the Template Store (see Chapter 5.1.3 page 79).

Media gallery: Enhancement of the CMS_INPUT_ CONTENTLIST input component enables several media to be selected from the project's Media Store and to be arranged and displayed within a gallery view (see Chapter 5.1.4 page 81).

Extension of the language-dependent inputs: In JavaClient on modelling workflows: The status, activities and transitions properties can now be assigned language-dependent display names and comments (see Chapter 5.1.5 page 83).
Analogous to the JavaClient, the language-dependent input options have now been extended in the server and project configuration too. A project's resolutions, language identifiers and template sets can now be assigned a language-dependent display name (see Chapter 6.1.1 page 101) and re-sorted (see Chapter 6.1.2 page 102).

WebEdit format templates for the "Easy-Edit" function: WebEdit 4.2 has been enhanced to include a new "Easy-Edit" function. The format templates required for this can be combined with the WebEdit format templates and are just like these are already included in the FirstSpirit scope of supply (see Chapter 5.1.6 page 85).

Templates: Simplified syntax for lambda expressions: The template language for lambda expressions has been simplified for FirstSpirit Version 4.2 (see Chapter 5.1.8 page 87).

Templates: New data element CMS_INCLUDE_OPTIONS: Using the new data element CMS_INCLUDE_OPTIONS the range of selectable values of input components offering several values can be configured dynamically (see Chapter 5.1.9 page 88).

Templates: Output of the menu name using the function MenuGroup: From FirstSpirit Version 4.2 the method *#nav.label* in the function MenuGroup has been changed (see Chapter 5.1.10 page 89).

New input components FS_REFERENCE and FS_DATASET: A new generation of input components is introduced with FirstSpirit Version 4.2, with the name prefix "FS_*" instead of "CMS_INPUT_*". These new input components will be gradually added within the scope of continued development of FirstSpirit and will be adjusted to clients' needs. For this reasons, the status of the new input components in Version 4.2 is "under development". They will not be officially released until FirstSpirit Version 5.0 (see Chapter 5.1.9 page 88).

GUI support for the definition of content areas: The "Properties" tab of a page template in the Template Store now contains input options for defining content areas and section restrictions (see Chapter 5.1.12 page 96).

Editing and deleting preview images: Preview images can now also be edited and deleted using the context menu “Extras“ (see Chapter 5.1.13 page 97).

2.4 For administrators

Extension of the language-dependent inputs: Analogous to the JavaClient, the language-dependent input options have now been extended in the server and project configuration too. A project's resolutions, language identifiers and template sets can now be assigned a language-dependent display name (see Chapter 6.1.1 page 101).

Order of resolutions, languages and template sets within the project: A project's resolutions, language identifiers and template sets can now be re-sorted within the server and project configuration (see Chapter 6.1.2 page 102).

Enhanced import function for database layer: The import function for projects can now be used to assign a new or existing database layer on the server to each individual schema of a project (see Chapter 6.1.3 page 102).

Project-wide pre-configuration of the Integrated preview: Administrators can define how the Integrated preview can be set by editors in a project in the project properties (see Chapter 6.1.4 page 104).

Enhanced FirstSpirit server log outputs: Cyclical output of the VM start time and the VM uptime (see Chapter 6.1.5 page 105).

2.5 Enhancements in FirstSpirit modules

A further area in which enhancements and add-ons are provided with FirstSpirit Version 4.2 is the FirstSpirit modules.

"FirstSpirit Enterprise Backup" module (data backup): New data backup strategies are introduced in FirstSpirit Version 4.2, which are specially tailored to the requirements of extensive enterprise projects:

- Shortest possible runtime
- Avoiding restrictions to editing operation
- Integration in commercial backup solutions

To this end, a new "Enterprise Backup" module is made available with FirstSpirit Version 4.2. The module enables efficient data backup using incremental and differential backups (see Chapter 6.2.1 page 105).

Notes about the use of this module, also in comparison with the archive function, can be found in the Chapter 6.2.2 page 107.

"FirstSpirit Office" module (Word import): The "FirstSpirit Office" module provides extensive infrastructure for importing MS Word documents into FirstSpirit input components. Initially, introduction of the module in FirstSpirit Version 4.2 covers the basic data acceptance requirements in the input components CMS_INPUT_DOM and CMS_INPUT_DOMTABLE. After installing the module, the FirstSpirit server and project configuration can be used to define rules for the conversion of CF HTML tags into FirstSpirit expressions. This definition can be made globally (server-wide) and if necessary can be overwritten for individual projects (see Chapter 5.2.1 page 98).

Licence-dependent "FirstSpirit remote media" function: The licence-dependent FirstSpirit remote media function has been enhanced for the input components CMS_INPUT_FILE, CMS_INPUT_PICTURE and CMS_INPUT_PAGEREF. Unlike FirstSpirit Version 4.1, in Version 4.2 it is now possible to define several remote projects (previously precisely one) for an input component. Analogous to the configuration of links, remote categories can now also be used within the CMS_INPUT_FILE, CMS_INPUT_PICTURE and CMS_INPUT_PAGEREF input components (see Chapter 5.2.2 page 99).

3 System requirements

This chapter explains the system requirements for FirstSpirit Version 4.2. The following details relate to the status at the time of the first release of FirstSpirit Version 4.2 and are not updated in this document.

For up-to-date and further information, please refer to the Technical Data Sheet for Version 4.2.

3.1 Operating systems

The following Java environments are supported at the time of releasing FirstSpirit Version 4.2:

FirstSpirit server

- Microsoft Windows 2000 Server on x86 32 bit
- Microsoft Windows 2003 Server on x86 32 bit, x64 edition
- Debian/GNU Linux 4.0 on x86 32 bit, AMD64/Intel64
- Debian/GNU Linux 5.0 on x86 32 bit, AMD64/Intel64
- Suse Linux Enterprise Server 9 on x86 32 bit, AMD64/Intel64
- Suse Linux Enterprise Server 10 on x86 32 bit, AMD64/Intel64
- Red Hat Enterprise Linux 4 on x86 32 bit, AMD64/Intel64
- Red Hat Enterprise Linux 5 on x86 32 bit, AMD64/Intel64
- Solaris 9 on Sparc 64Bit
- Solaris 10 on Sparc 64bit and AMD64/Intel64
- AIX 5.3L 64bit

Note on 64Bit x86 CPU architectures: Only AMD64 and the compatible Intel64, also known as EMT64, are supported. No JDK 1.5 and 1.6 is offered for Intel's outdated IA64. It is therefore not supported by FirstSpirit.

FirstSpirit client

- Microsoft Windows XP
- Microsoft Windows Vista

Notes about the operation under MacOS X and Linux: The FirstSpirit JavaClient is in principle developed in a way that it can be operated under Mac-OS X and Linux. Because of the considerable differences compared to windows-based systems (especially concerning the frontend libraries) problems can occur in detail. They manifest themselves normally only in slight variances in the representation or in very particular functions (e.g. drag & drop between JavaClient and operating system). Malfunctions under MacOS X and Linux due to the operating system can not be excluded in spite of periodical quality assurance. Such malfunctions will be eliminated, as far as technically possible, within the scope of the software maintenance.

3.2 Java environment

The following Java environments are supported at the time of releasing FirstSpirit Version 4.2:

FirstSpirit server

- Sun Java 6 (JDK1.6), both in the 32bit and in the 64bit version under Windows and Linux, is recommended, as well as Solaris (on Sparc, 64bit only) and IBM JDK 1.5 on AIX (64bit only)

FirstSpirit client

- Sun Java 6 (JRE1.6), 32 bit version with Java Web Start is recommended.

Basically, it is the aim of the FirstSpirit development to support both the current Sun JDK and the preceding JDK. However, at the time of releasing FirstSpirit Version 4.2, this is problematic as the JDK-1.5 line is from 2004 and many areas of it no longer match state of the art standards. The next Sun Open JDK-7 has also not yet been officially released. Therefore the release policy for FirstSpirit Version 4.2 is defined as follows with regard to JDK support:

Note about the usage of Sun JDK 1.5:

- Sun JDK 1.5 is NO LONGER supported for the FirstSpirit server.
- Sun JDK 1.5 is now only supported to a limited extent for FirstSpirit JavaClient. This means it is still possible to use FirstSpirit JavaClient with JDK 1.5, however, regular quality assurance will no longer be carried out in this configuration. It is therefore possible for problems to occur under JDK 1.5, which no longer exist in JDK 1.6. If such JDK-1.5 specific problems occur while running FirstSpirit JavaClient, the client must first change over to JDK 1.6. **Support for JDK-specific problems is no longer possible for Sun Java 5 (JDK 1.5) under FirstSpirit Version 4.2.**

With the official release of Sun Open JDK 7, we will try to support this JDK as quickly as possible (primarily for FirstSpirit server).

3.2.1 Known JDK problems

FirstSpirit is released for a main version of a JDK, e.g. Sun Java6 (JDK1.6) or IBM JDK 1.5. Regular updates of JDK are issued as part of the debugging in JDK; these updates are checked within the scope of quality assurance to ensure they are compatible with FirstSpirit. In very rare cases, incompatibilities can occur due to errors in JDK, which cannot be handled in programming terms. This is not a FirstSpirit error! In this case we recommend falling back on an older JDK version or – if available – using a newer JDK version. Please contact the FirstSpirit Helpdesk if you have any questions about known problems:

- Editorial permissions can only be edited via FirstSpirit JavaClient from a Java version higher than 1.5.0_05. Errors occur in older JREs.
- Because of a bug of the signature check in Java Web Start for the **FirstSpirit JavaClient** the following versions of Sun Java Web Start (part of the JRE) can not be used: 1.6.0_14, 1.6.0_15 and 1.6.0_16. The bug has been fixed in the version 1.6.0_17 by SUN, but this version can not yet be released for the use with FirstSpirit. For this reason, Java Web Start **1.6.0_13 or older** should be used for the FirstSpirit JavaClient. Java Web Start 1.6.0_13 or older can be combined with a different version of the JRE by configuration of the operating system or of the web browser
- Because of a bug of the Java Hotspot Compiler the following JDK versions can not be used for the **FirstSpirit server**: 1.6.0_10, 1.6.0_11, 1.6.0_12 and 1.6.0_13. For this reason, the version **1.6.0_14 or newer** should be used for the FirstSpirit server.

> ! *For more information about the choice of the JDK see also FirstSpirit Installation Instructions (Version 4.2), Chapter 3.2 "Java environment selection" (for Windows) and Chapter 4.4 "Java" (for GNU/Linux, Solaris or AIX).*

3.3 Databases

The following databases are supported at the time of releasing FirstSpirit Version 4.2:

MySQL:

- MySQL 4.1 (Linux and Windows)
- MySQL 5.0 (Linux and Windows)

Microsoft SQL Server

- Microsoft SQL Server 2000 (for Windows servers only)
- Microsoft SQL Server 2005 (for Windows servers only)

Oracle

- Oracle 9 (Solaris and Linux)
- Oracle 10 (Solaris, Linux and Windows)

IBM DB2

- IBM DB2 8.2 (Linux and AIX)
- IBM DB2 9.1 (Linux, AIX and Windows)

PostgreSQL

- PostgreSQL 8.0 (Linux and Windows)
- PostgreSQL 8.2 (Linux and Windows)

3.4 Browsers

Both FirstSpirit WebClient and Server Monitoring are purely web-based applications, which set some very high requirements regarding the web browser used.

The following browsers are supported at the time of releasing FirstSpirit Version 4.2:

- Microsoft Internet Explorer 6.0 and 7.0
- Mozilla Firefox 2.0

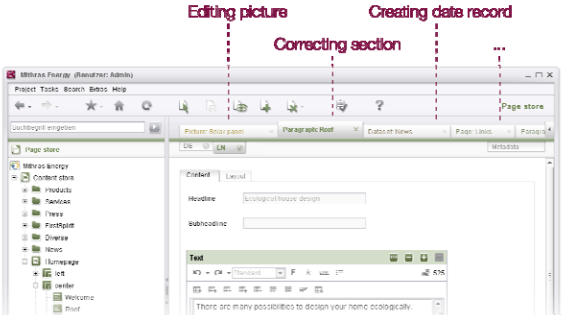
4 New/modified functions for editors

4.1 New functions in JavaClient

4.1.1 Editing in several workspaces ("Multi-tabbing")

One of the most conspicuous changes for editors is the introduction of tab navigation in the editing area of FirstSpirit JavaClient.

The user can use the tabs to quickly navigate between several project nodes (for example, individual pages, sections or media), without selecting the required objects again in the tree view. Once opened, the different workspaces (e.g. sections) are kept in the editing area as tabs, until they are explicitly closed by the modifier. This so-called "multi-tabbing" enables users to very conveniently and clearly **edit several workspaces**.



Headline: Ecological house design

There are many possibilities to design your home ecologically.

Figure 4-1: Multi-Tabbing

While, for example, a dataset is edited in one workspace, parallel to this, a picture can be changed in another workspace.

In this way the editor can set up their own **personalised workspaces** which ideally suit their tasks. The respective focus is retained when the tab is changed, so that the user is always displayed the last edited area in the respective tab (for example, a specific input component within a form).

It is also possible to switch from the active workspace to the adjacent workspace using **keyboard shortcuts**:

- Ctrl + TAB: Select workspace to the right, next to the active workspace.
- Ctrl + Shift + TAB: Select workspace to the left, next to the active workspace.

An element in a workspace can either be opened using the active workspace or the element can be opened directly in a new workspace using the middle mouse button or using the context menu. A workspace can be fixed using the context menu, to prevent an element in the active workspace from being lost when a new element is selected. **Fixed workspaces** are labelled with a paper clip and are retained for all actions, until they are explicitly closed by the editor:

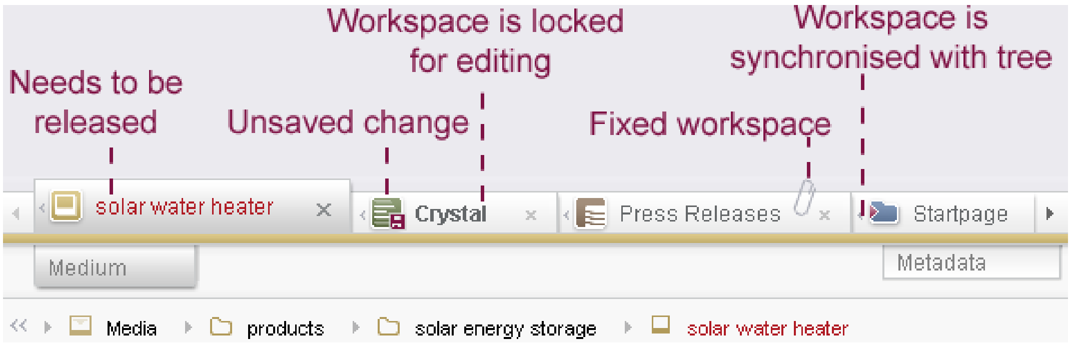


Figure 4-2: Display of fixed and unfixed workspaces

Apart from the name and type of open element, the tabs contain other information, for example the change status.

The **change status of an element** is denoted by three colours:

- Black: Element has been released
- Red: Element has been changed, the changes have not yet been released
- Blue: A workflow has been started on the element.

The colour coding for the change status of an element has been limited from the previous individually configurable colour coding to the three colours introduced above. From FirstSpirit Version 5.0, the language-dependent enhancements should be increasingly shown for the status of an element in the tree in order to clearly demarcate statuses of different workflows from each other, for example: Other pages [waits for release] (cf. Chapter 5.1.5 page 83).

Apart from the change status (colour of the lettering) the display also visually shows whether an element in an open workspace is locked to prevent editing (lettering is bold) or not and whether unsaved changes exist in a workspace (see Figure 4-2).

The individual workspaces can be moved within the tab area by means of Drag & Drop.

Apart from editing in individual workspaces, it is also possible to **Drag & Drop between the individual workspaces**. For example, media from the thumbnail view of a workspace can be copied into a picture input component of another workspace by means of Drag & Drop (see Chapter 4.1.10 page 45).

The workspaces are closed by using the icon ✕ of the respective workspace, using the context menu's item **Close workspace** or the keyboard shortcut Ctrl + W. If only one workspace is to be remained open the other workspaces can be closed by clicking on the item **Close other workspaces**.

The "**Restore Closed Workspaces**" function can be used to re-open already closed workspaces. The function can be invoked using the context menu on each tab of an open workspace (or using the "View" menu, see Figure 4-4) and shows the ten most recently closed workspaces of the current user session.

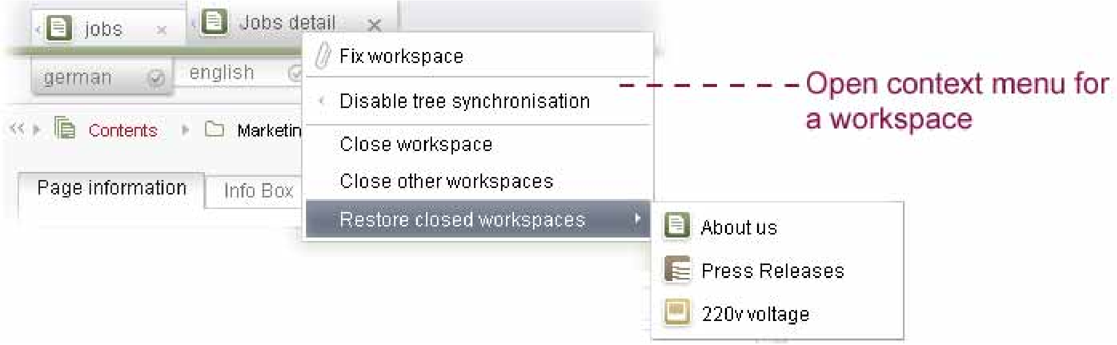


Figure 4-3: Context menu on a workspace

The whole user-specific workspace (with all open tabs) can be saved using the "View" menu (see Figure 4-4), either at defined points in time (**"Save Open Workspaces"**) or on exiting JavaClient (**"Save on Exiting"**), in order to be able to restore the required areas again the next time the JavaClient is started (**"Restore Settings on Restart"– "Workspaces"**) (see Chapter 8.1.7 page 117).

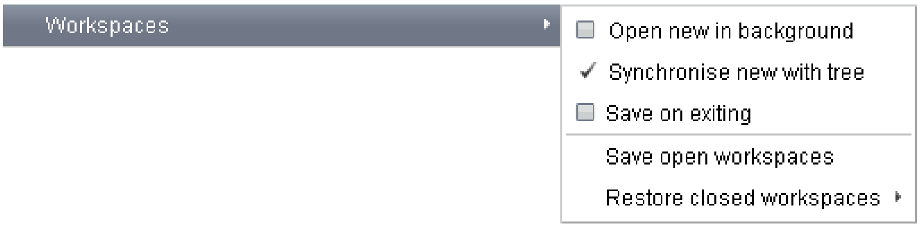
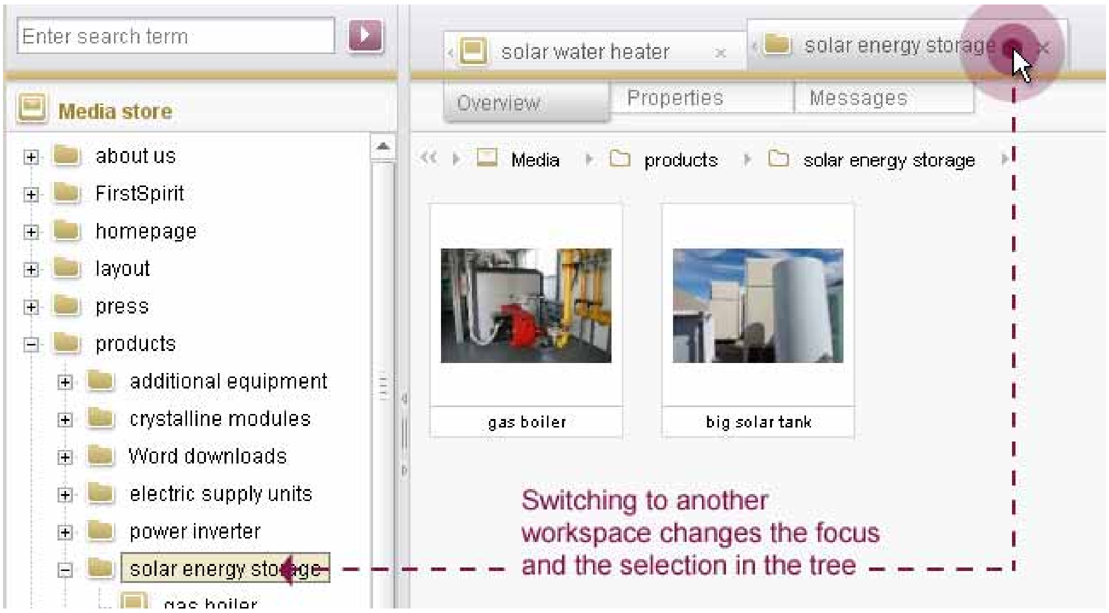


Figure 4-4: "View" menu – Workspaces

Working with workspaces can be configured for the editor – both globally for the whole JavaClient using the new "View" menu (see Figure 4-4) as well as locally for each individual workspace using the context menu of the respective tab (see Figure 4-3). For example, the editor can use the "View" menu to define whether to **"Open New (Workspaces) in the Background"** or whether to fetch them directly into the foreground as an active workspace.

The **"Synchronise New (Workspaces) with Tree"** function (in the "View" menu, see Figure 4-4) is helpful for editors who have worked a lot with FirstSpirit's tree navigation to date. The active workspace in the right-hand part of the FirstSpirit JavaClient is automatically synchronised with the tree view in the left-hand workspace. Switching tabs in the right-hand editing area therefore has a direct effect on the tree navigation in the left-hand part of JavaClient:



Switching to another workspace changes the focus and the selection in the tree

Figure 4-5: Tree synchronisation on changing a workspace

The synchronisation can also be individually enabled or disabled for each workspace using the tab's context menu (**"Enable/Disable Tree Synchronisation",** see Figure 4-3). In this way, synchronisation with the tree can be helpful, for example, when editing pages and sections in the Page Store, whereas it will tend not to be needed for editing several datasets in a table view in the Content Store.

The **Selection Model in the Tree** has also been changed over within the scope of this synchronisation:

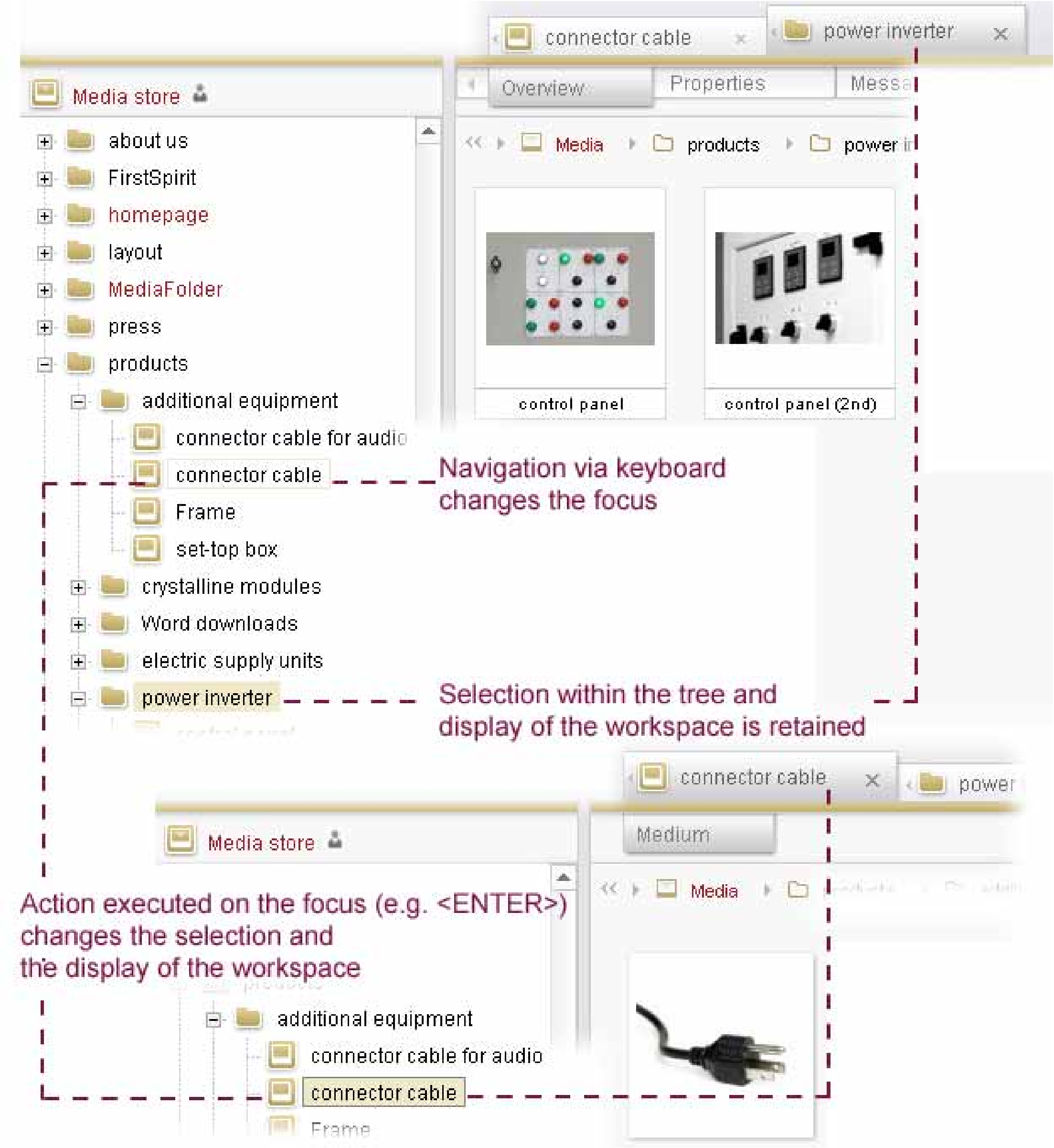


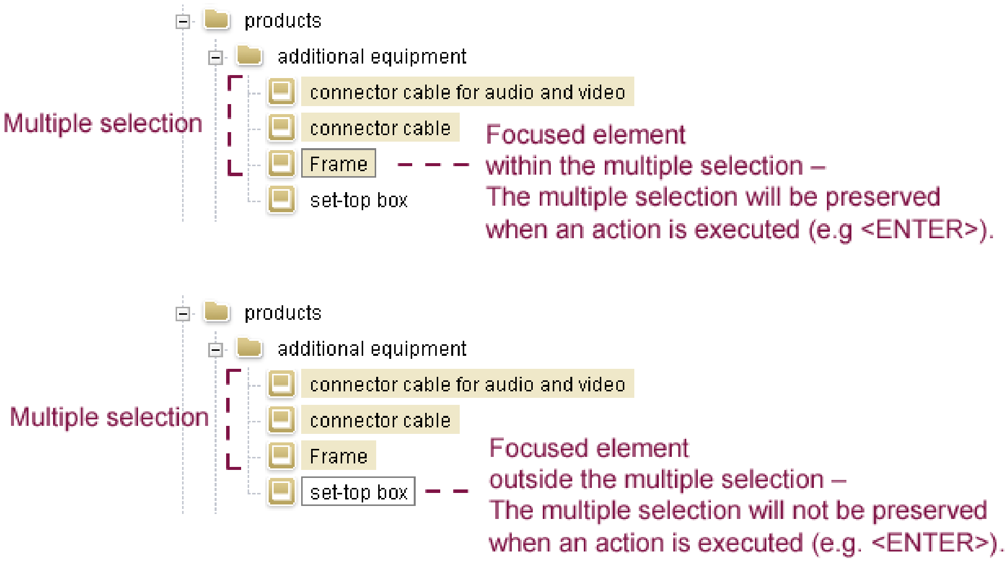
Figure 4-6: Selection and focussing during navigation using the keyboard

- Navigation using the keyboard: If the user navigates using the keyboard (within the tree view), the current selection (visualised by a coloured background) in the workspace and in the tree is initially retained; only the focus within the tree view (visualised by a coloured frame) changes. The active workspace is therefore not affected by navigation through the tree elements using the keyboard. Only when an action takes place on the focussed element (e.g. Ctrl + V) or the element is explicitly selected by clicking <ENTER>, does the selection change and the element is opened within the active workspace:
- Multiple selection: The following applies to multiple selection within the tree: All selected elements are assigned a background colour, which depends on the respective Store. The last selected element is also focussed and therefore has a

black frame. If the focussed element lies outside the multiple selection (e.g. in the case of navigation using the keyboard) it is marked by a coloured frame (see Figure 4-7).

Multiple selection using the keyboard is also possible (Shift + ↑ or Shift + ↓). Here, analogous to "Navigation using the Keyboard", the focus is changed first (coloured frame). The elements of the multiple selection are not selected (and highlighted with a coloured background) until <ENTER> is clicked or an action is performed.

The active workspace is not changed by a multiple selection. Unlike individual selection, in most cases display of all elements of a selection is not wanted. If the elements are nevertheless to be opened in workspaces, the context function "Open in New Workspace" can be used on the multiple selection.



Focused element within the multiple selection – The multiple selection will be preserved when an action is executed (e.g <ENTER>).

Focused element outside the multiple selection – The multiple selection will not be preserved when an action is executed (e.g. <ENTER>).

Figure 4-7: Multiple selection in the tree

On switching the workspace (with enabled tree synchronisation), an existing multiple selection in the tree:

- is retained if the element in the workspace is part of the multiple selection.
- is not retained if the element in the workspace is part of the multiple selection.

In later versions of FirstSpirit, the tab navigation concept could be extended, for example, in order to display the results of a search, the task list of an editor or the version history of a project node within workspaces.

The "multi-tabbing" function is purely a JavaClient function and is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.2 Breadcrumb navigation

Another innovation is that the path from the Store root up to the current element is displayed above the form area in the form of a "breadcrumb navigation". This display is particularly helpful if synchronisation of the workspace with the tree is not enabled (see Figure 4-4).

Apart from pure information content, the path elements can also be used to navigate to other adjacent elements on the respective level. At the same time, the currently open element is highlighted (see Figure 4-8).

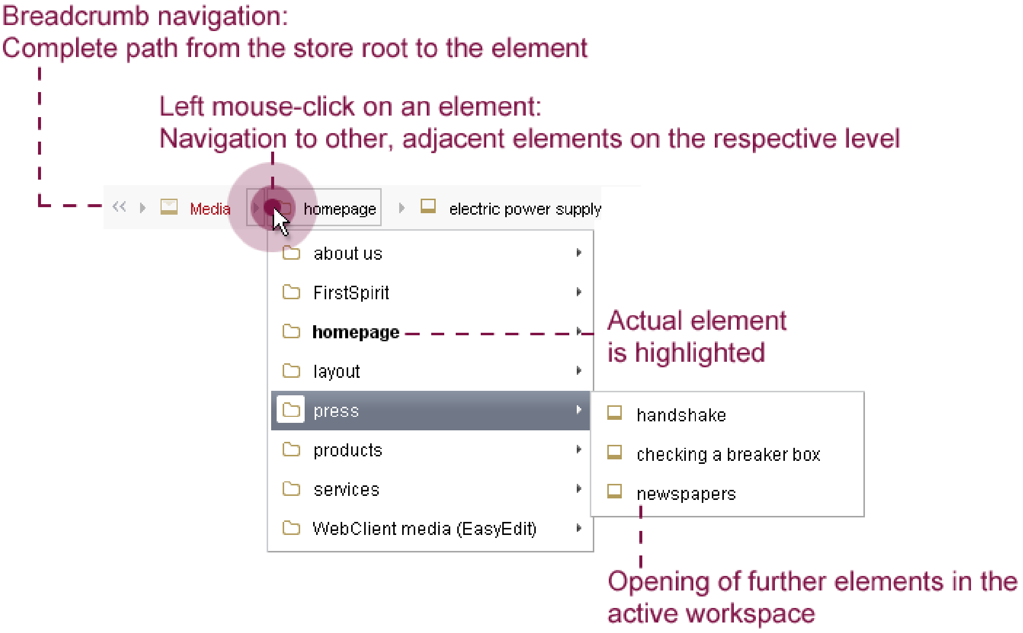

Breadcrumb navigation:
Complete path from the store root to the element
Left mouse-click on an element:
Navigation to other, adjacent elements on the respective level
Media
homepage
electric power supply
about us
FirstSpirit
homepage
layout
press
products
services
WebClient media (EasyEdit)
handshake
checking a breaker box
newspapers
Actual element
is highlighted
Opening of further elements in the
active workspace

Figure 4-8: Navigation using the path elements ("breadcrumb navigation")

Analogous to opening the context menu in the tree, the menu can also be opened on a path element, for example to start a release on the page which has just been edited (see Figure 4-9).

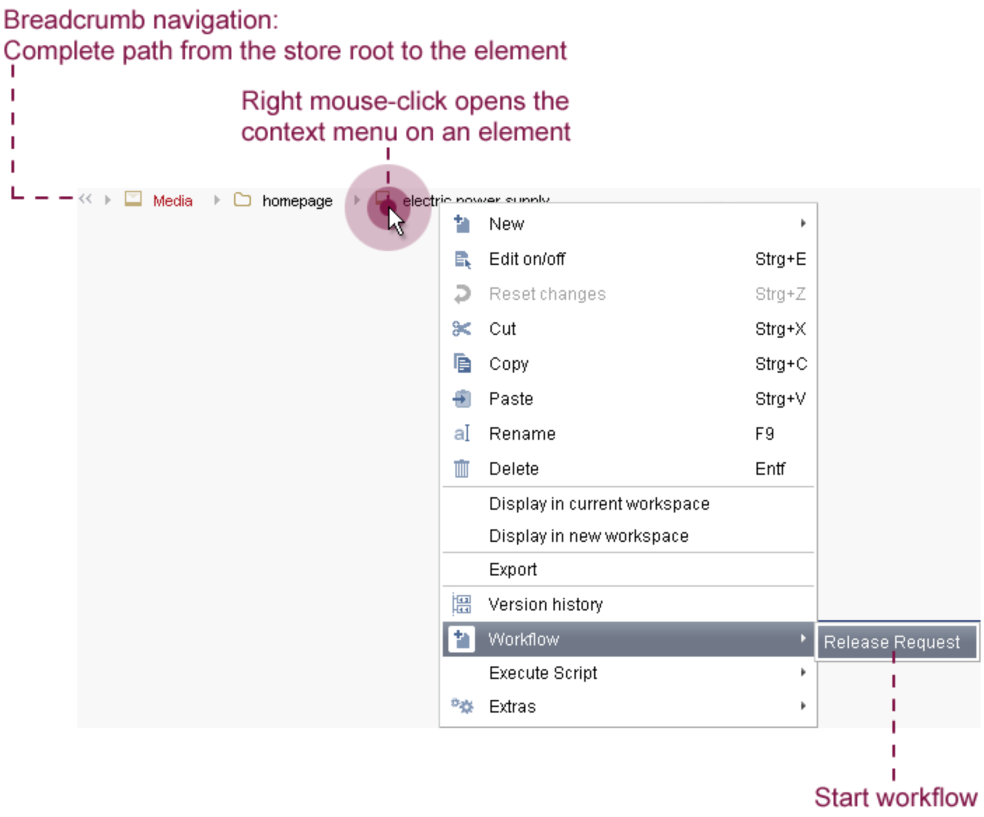


Figure 4-9: Opening the context menu on a path element

The colour coding for visualisation of an object's change status (black, red, blue) is, analogous to the tab lettering of the workspaces, also used within the breadcrumb navigation (see Chapter 4.1.1 page 17).

The "breadcrumb navigation" function is purely a JavaClient function and is ***not*** *supported in WebEdit (see Chapter 4.3 page 74).*

4.1.3 Display of the Stores as navigation areas

Within the scope of introduction of "multi-tabbing", the overall project display in the left-hand navigation area of FirstSpirit JavaClient is also to be replaced by clearer individual display of the Stores. At the same time, only one store is to be displayed as a tree in the navigation area of JavaClient. The other Stores are displayed as space-saving buttons below the tree display. If the user rarely uses individual Stores or does not use them at all, they can also be hidden or displayed as a small icon only in the footer bar (see Figure 4-10). When a button is clicked, the required store is shown as a navigation tree and the area open to date is hidden again.

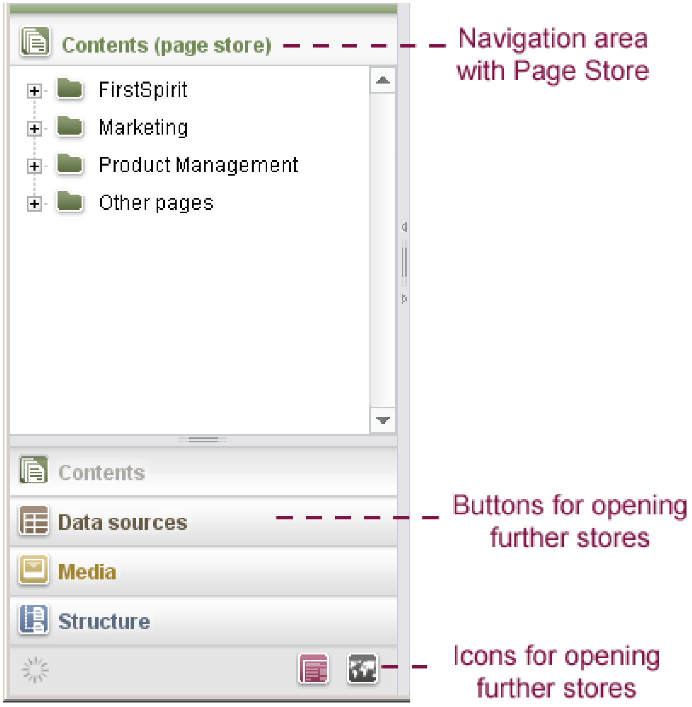


Figure 4-10: New navigation in the Store

Analogous to tab navigation in the right-hand editing area of JavaClient ("multi-tabbing"), here too the focus within the individual navigation areas is retained For example, if the user edits a certain page within the Content Store, the focus remains on the page, even if the user switches to the Media Store to select a medium and then switches back into the Content Store again. Scrolling and searching between different Stores within the project tree display used to date is therefore no longer necessary. It is, of course, still possible to Drag & Drop across several Stores.

Individual display of the stores as navigations areas is purely a JavaClient function and is not supported in WebEdit (see Chapter 4.3 page 74).

4.1.4 Integrated preview in JavaClient

The integrated preview provides the editor with a direct WYSIWYG preview of a page within FirstSpirit JavaClient. Apart from the form-based workspace in JavaClient, at the same time the content of the page is displayed in the integrated browser.

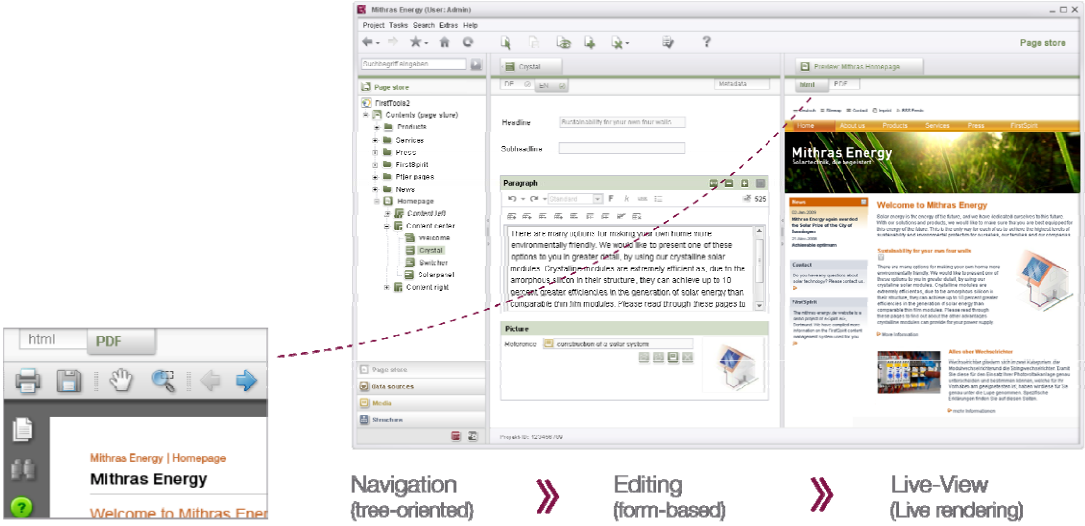


Figure 4-11: Display of the WYSIWYG preview

All editorial changes are automatically updated in the preview area (after saving). The **changeover between the various output channels** takes place simply via tabs within the integrated preview. Elaborate switching between the workspace in JavaClient and the preview in the web browser is therefore no longer necessary.

The integrated preview can be used in different stores. Within the Content and Site Store, the editorial content is displayed in the respective output channel, for example as an HTML page or as a PDF (see Figure 4-12).

Note: The display of an output channel in the integrated preview can possibly require the installation of further plug-ins. For example, the display of an integrated preview in the output channel PDF the Acrobat Reader plug-in must be installed and configured correctly on the workstation computer of the editor.

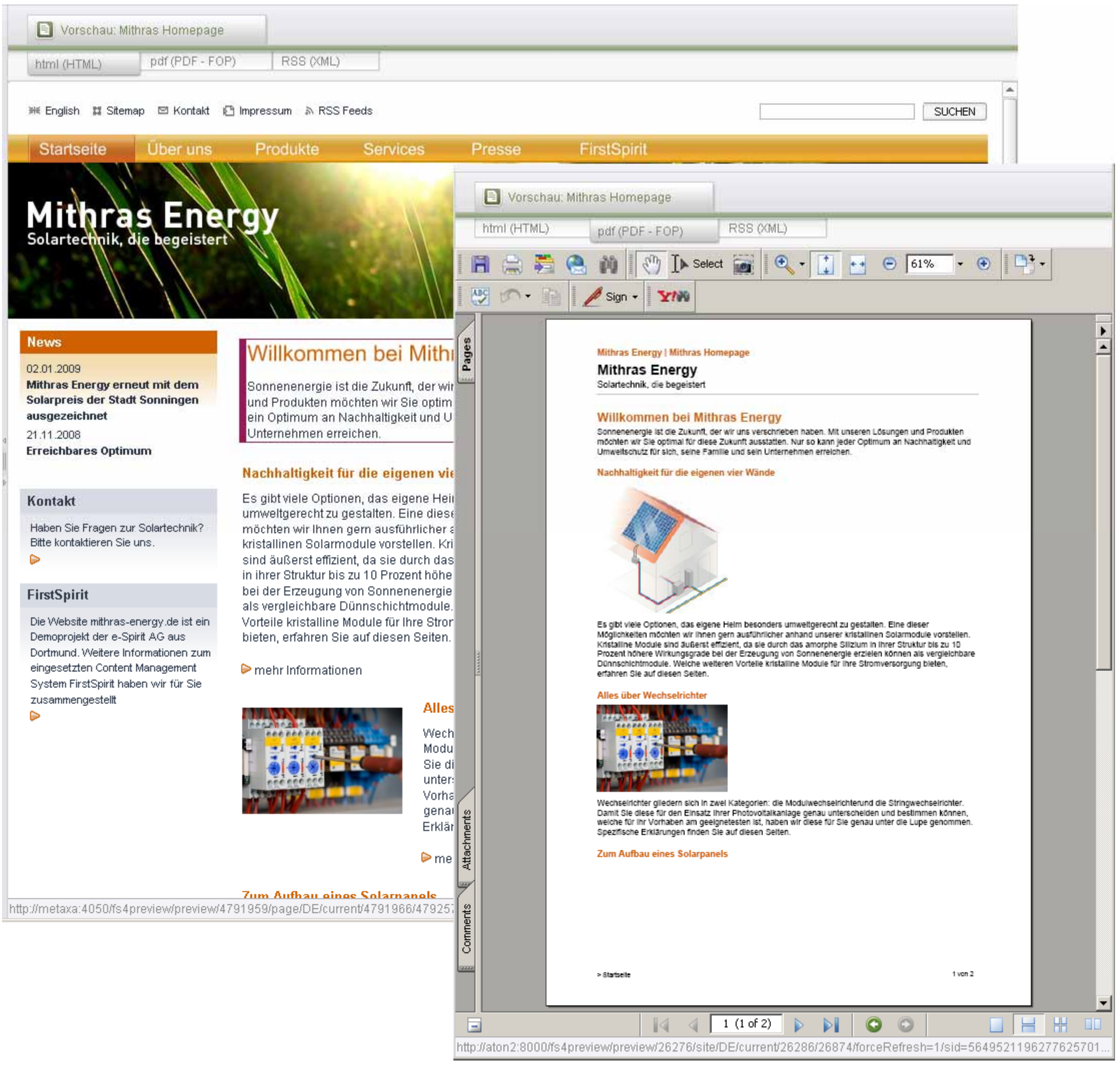
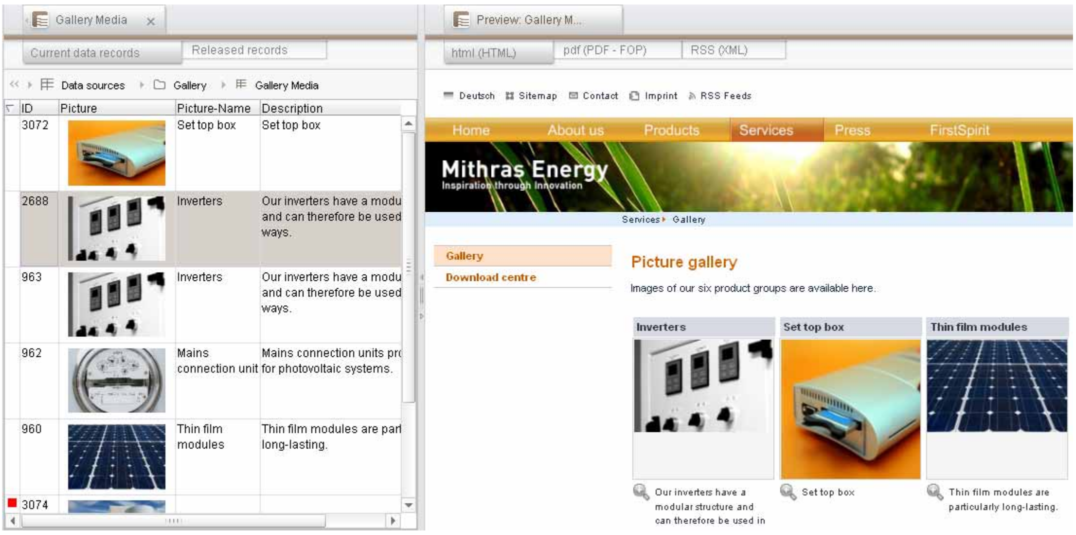


Figure 4-12: Integrated preview in the HTML and PDF output channel

Representations of the datasets can be displayed within the page context in the Content Store. If a certain dataset is selected in FirstSpirit's Content Store, use of the dataset from the Page Store is displayed in the integrated preview.



ID	Picture	Picture-Name	Description
3072		Set top box	Set top box
2688		Inverters	Our inverters have a modu and can therefore be used ways.
963		Inverters	Our inverters have a modu and can therefore be used ways.
962		Mains connection unit	Mains connection units pro for photovoltaic systems.
960		Thin film modules	Thin film modules are part long-lasting.
3074			

Picture gallery

Images of our six product groups are available here.

Our inverters have a modular structure and can therefore be used in

Set top box

Thin film modules are particularly long-lasting.

Figure 4-13: Representation of the dataset in the integrated preview

A **context menu** with the following functions can be opened within the integrated preview:

- Reload
- Next/Back
- Display page source text (keyboard shortcut: Ctrl + Shift + U)

The **integrated preview is configured** using the new "View" menu (cf. Chapter 8.1.7 page 117). The integrated preview can be optionally displayed on the right next to the workspace or, on smaller monitors, in a separate window (note: The "Content Highlighting" function is not supported if the integrated preview is displayed in an external window – cf. Chapter 4.1.5).

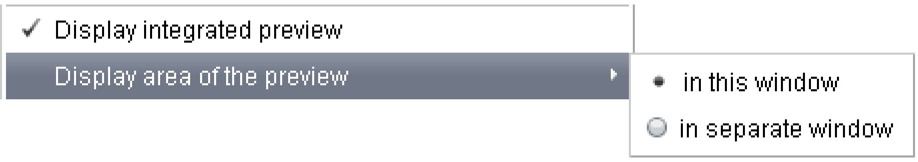


Figure 4-14: "View" menu – Configure integrated preview

In addition, a scaling factor of 50% to 200% for the HTML output channel can be defined in the bottom part of the preview (see Figure 4-15). This means it is possible to scale up or down the size of the content shown in JavaClient's preview area. The set scaling factor is saved as a user-specific setting and is retained until it is next changed.

Scaling can only be executed in the HTML output channel.

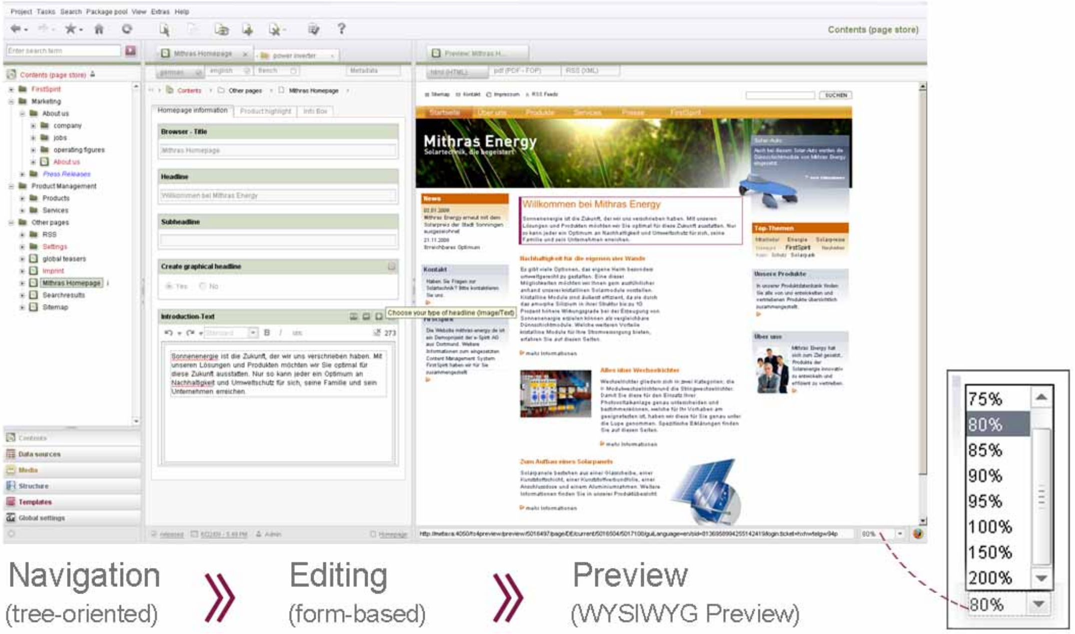


Figure 4-15: Setting the scaling factor for the integrated preview

The integrated preview can also be used within the Template Store (see Chapter 5.1 page 76).

Note: It is not possible to guarantee that all plug-ins can be used within the integrated preview. Common plug-ins, such as Adobe Flash or Adobe Acrobat should however not lead to problems within the preview.

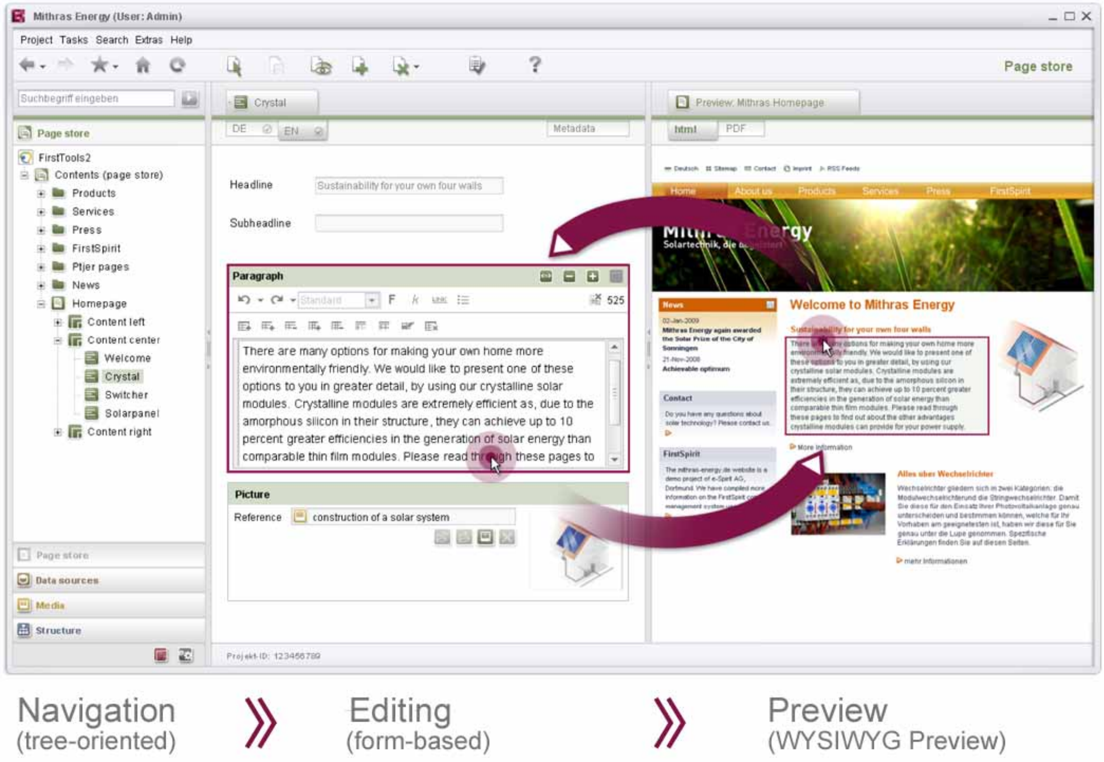
Note: The display of Java Applets and display of HTML tool tips is not supported within the integrated preview.

Note: The “integrated preview” function in FirstSpirit JavaClient bases on the Mozilla browser engine XUL-Runner. At the moment, it is not available as 64bit version. The requirement for the use of the preview integration is thus a 32bit JRE/JDK. The browser integration is not compatible with a 64 bit JRE/JDK.

Note: The “integrated preview” function can be used with the operating systems MacOS X and Linux only to a very limited extent at the moment. The manufacturer of the third-party software, used by FirstSpirit, is already aware of the problems and works on bugfixing. For the time being, we advise not to use the integrated preview under MacOS X and Linux until a debugging is available and has been tested.

The "integrated preview" function is purely a JavaClient function and is not supported in WebEdit (see Chapter 4.3 page 74).

4.1.5 Content Highlighting



Headline: Sustainability for your own four walls

Subheadline:

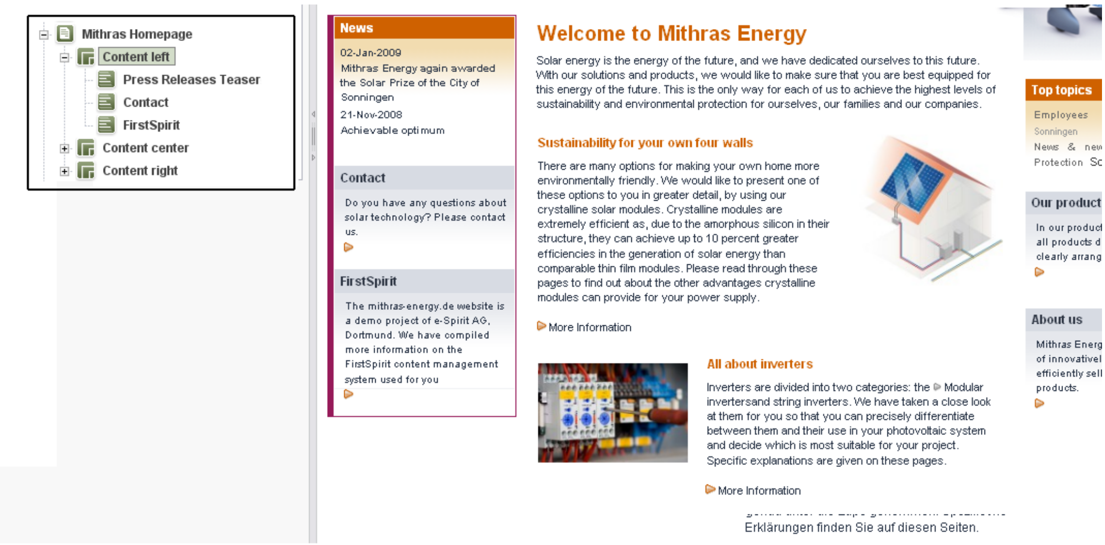
There are many options for making your own home more environmentally friendly. We would like to present one of these options to you in greater detail, by using our crystalline solar modules. Crystalline modules are extremely efficient as, due to the amorphous silicon in their structure, they can achieve up to 10 percent greater efficiencies in the generation of solar energy than comparable thin film modules. Please read through these pages to

Reference: construction of a solar system

Figure 4-16: Content Highlighting – Find content with a click

Apart from integrated preview, "**Content Highlighting**" provides further help to the editor. If the editor is in the form-based workspace of the JavaClient, for example within an input component, the currently edited area is highlighted in colour in the preview. The editor can therefore trace at any time, how and where the changes just made affect the page. But "Content Highlighting" works the other way around: If the editor clicks an element in the preview, for example, a text passage, the corresponding section is automatically opened in the workspace and the input component is highlighted in the form area. Search and navigation for specific content in JavaClient is therefore no longer necessary. Texts and pictures can be quickly and easily found and changed.

The highlighted information depends on the context in which it is opened. Apart from individual input components it is possible, for example, to highlight the complete page information in the integrated preview by clicking a page. On the other hand, if a content area or a section is selected, only the content of the content area or of the section is outlined with a frame.



News

02-Jan-2009
Mithras Energy again awarded the Solar Prize of the City of Sonningen

21-Nov-2008
Achievable optimum

Contact

Do you have any questions about solar technology? Please contact us.

FirstSpirit

The mithras-energy.de website is a demo project of e-Spirit AG, Dortmund. We have compiled more information on the FirstSpirit content management system used for you

Welcome to Mithras Energy

Solar energy is the energy of the future, and we have dedicated ourselves to this future. With our solutions and products, we would like to make sure that you are best equipped for this energy of the future. This is the only way for each of us to achieve the highest levels of sustainability and environmental protection for ourselves, our families and our companies.

Sustainability for your own four walls

There are many options for making your own home more environmentally friendly. We would like to present one of these options to you in greater detail, by using our crystalline solar modules. Crystalline modules are extremely efficient as, due to the amorphous silicon in their structure, they can achieve up to 10 percent greater efficiencies in the generation of solar energy than comparable thin film modules. Please read through these pages to find out about the other advantages crystalline modules can provide for your power supply.

More Information

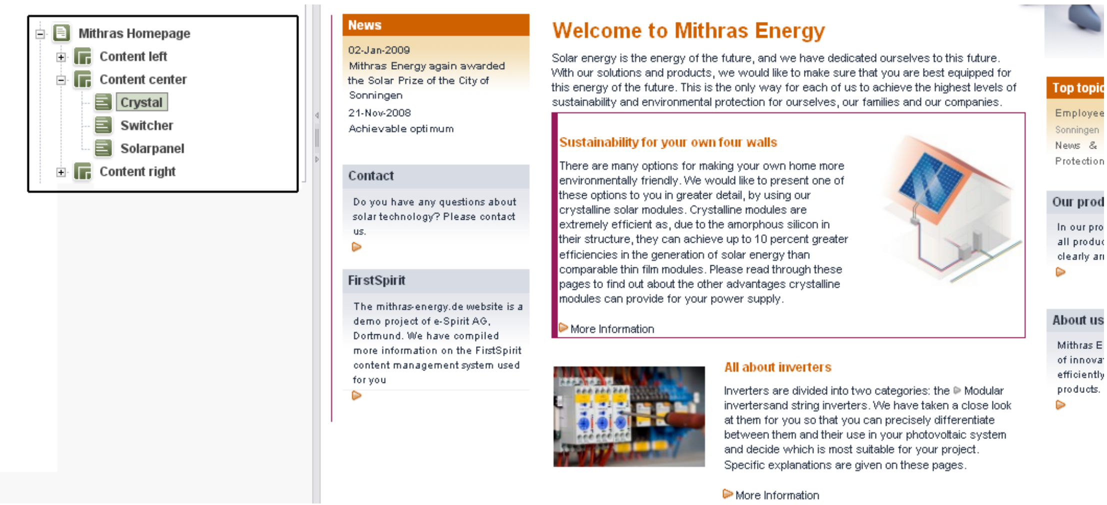
All about inverters

Inverters are divided into two categories: the Modular invertersand string inverters. We have taken a close look at them for you so that you can precisely differentiate between them and their use in your photovoltaic system and decide which is most suitable for your project. Specific explanations are given on these pages.

More Information

Erklärungen finden Sie auf diesen Seiten.

Figure 4-17: Content Highlighting in the integrated preview – Content area



News

02-Jan-2009
Mithras Energy again awarded the Solar Prize of the City of Sonningen

21-Nov-2008
Achievable optimum

Contact

Do you have any questions about solar technology? Please contact us.

FirstSpirit

The mithras-energy.de website is a demo project of e-Spirit AG, Dortmund. We have compiled more information on the FirstSpirit content management system used for you

Welcome to Mithras Energy

Solar energy is the energy of the future, and we have dedicated ourselves to this future. With our solutions and products, we would like to make sure that you are best equipped for this energy of the future. This is the only way for each of us to achieve the highest levels of sustainability and environmental protection for ourselves, our families and our companies.

Sustainability for your own four walls

There are many options for making your own home more environmentally friendly. We would like to present one of these options to you in greater detail, by using our crystalline solar modules. Crystalline modules are extremely efficient as, due to the amorphous silicon in their structure, they can achieve up to 10 percent greater efficiencies in the generation of solar energy than comparable thin film modules. Please read through these pages to find out about the other advantages crystalline modules can provide for your power supply.

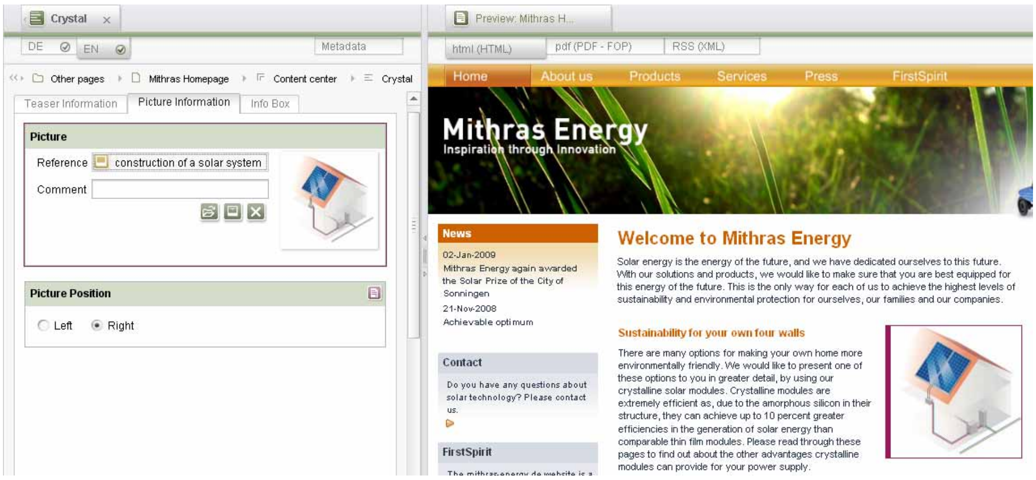
More Information

All about inverters

Inverters are divided into two categories: the Modular invertersand string inverters. We have taken a close look at them for you so that you can precisely differentiate between them and their use in your photovoltaic system and decide which is most suitable for your project. Specific explanations are given on these pages.

More Information

Figure 4-18: Content Highlighting in the integrated preview – Section area



Reference: construction of a solar system

Comment:

○ Left ● Right

02-Jan-2009
Mithras Energy again awarded the Solar Prize of the City of Sonningen
21-Nov-2008
Achievable optimum

Do you have any questions about solar technology? Please contact us.

The mithras-energy.de website is a

Welcome to Mithras Energy

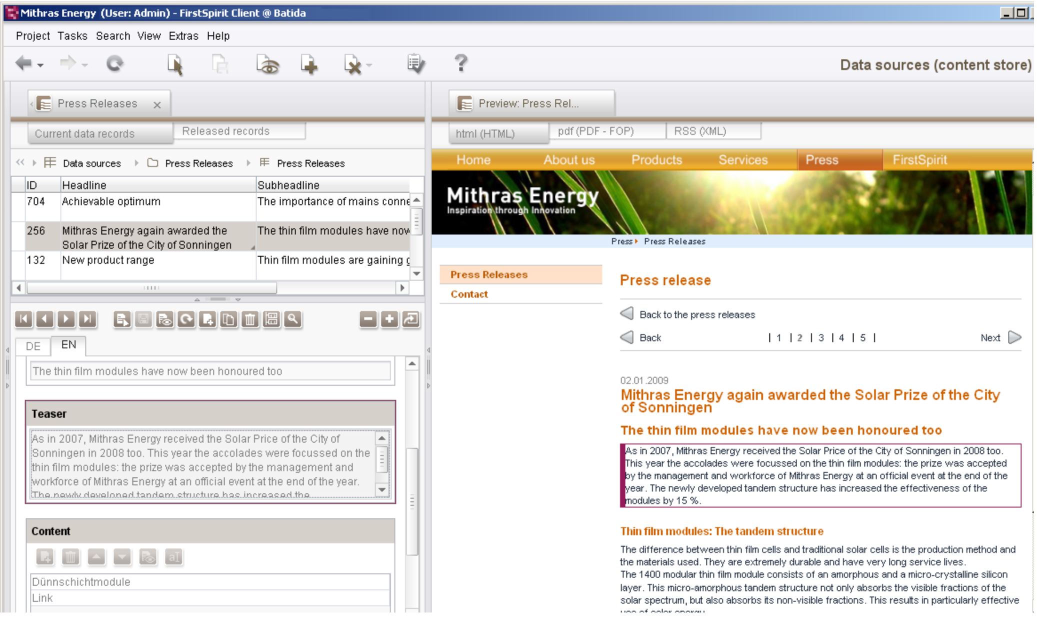
Solar energy is the energy of the future, and we have dedicated ourselves to this future. With our solutions and products, we would like to make sure that you are best equipped for this energy of the future. This is the only way for each of us to achieve the highest levels of sustainability and environmental protection for ourselves, our families and our companies.

Sustainability for your own four walls

There are many options for making your own home more environmentally friendly. We would like to present one of these options to you in greater detail, by using our crystalline solar modules. Crystalline modules are extremely efficient as, due to the amorphous silicon in their structure, they can achieve up to 10 percent greater efficiencies in the generation of solar energy than comparable thin film modules. Please read through these pages to find out about the other advantages crystalline modules can provide for your power supply.

Figure 4-19: Content Highlighting of a picture input component

"Content Highlighting" also works for structured content from the Content Store. To this end, a dataset can be simply marked within the Content Store. A representation of the dataset is then displayed within a page context in the preview. If the dataset is edited, when an input component is clicked, the content of the input component is highlighted in the preview.



ID	Headline	Subheadline
704	Achievable optimum	The importance of mains conne
256	Mithras Energy again awarded the Solar Prize of the City of Sonningen	The thin film modules have now
132	New product range	Thin film modules are gaining g

The thin film modules have now been honoured too

As in 2007, Mithras Energy received the Solar Price of the City of Sonningen in 2008 too. This year the accolades were focussed on the thin film modules: the prize was accepted by the management and workforce of Mithras Energy at an official event at the end of the year. The newly developed tandem structure has increased the

Dünnschichtmodule
Link

Press release

Back to the press releases

Back | 1 | 2 | 3 | 4 | 5 | Next

02.01.2009
Mithras Energy again awarded the Solar Prize of the City of Sonningen

The thin film modules have now been honoured too

As in 2007, Mithras Energy received the Solar Price of the City of Sonningen in 2008 too. This year the accolades were focussed on the thin film modules: the prize was accepted by the management and workforce of Mithras Energy at an official event at the end of the year. The newly developed tandem structure has increased the effectiveness of the modules by 15 %.

Thin film modules: The tandem structure

The difference between thin film cells and traditional solar cells is the production method and the materials used. They are extremely durable and have very long service lives.
The 1400 modular thin film module consists of an amorphous and a micro-crystalline silicon layer. This micro-amorphous tandem structure not only absorbs the visible fractions of the solar spectrum, but also absorbs its non-visible fractions. This results in particularly effective use of solar energy.

Figure 4-20: Content Highlighting the input components of a dataset

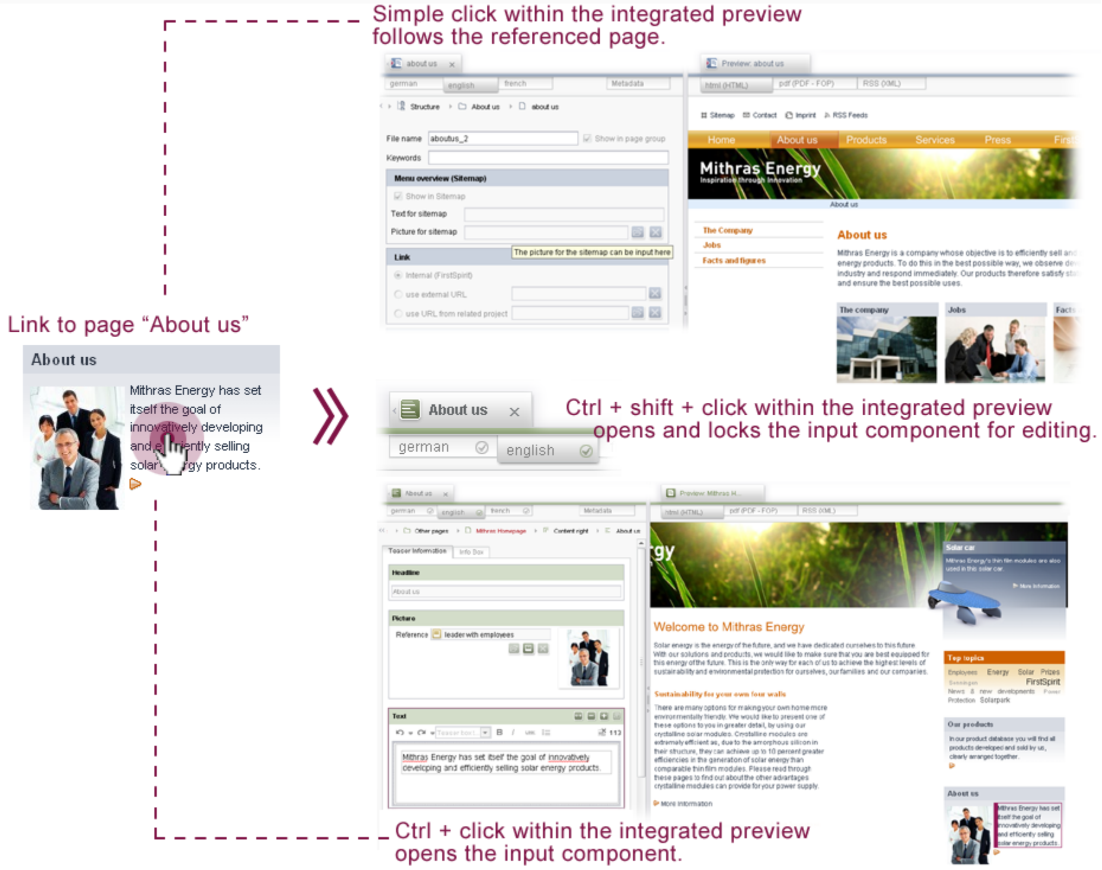
Conversely, content can also be selected within the preview. If a text or picture display within the integrated preview is clicked, the corresponding form opens directly in the editing area.

Keyboard shortcuts or the context menu can be used to open other actions on the highlighted object (within the integrated preview):

- **Click in the integrated preview:** If the "Workspace follows Preview" option is enabled, the workspace switches to the corresponding object in JavaClient.
- **Shift + click in the integrated preview:** If the "Workspace follows Preview" option is enabled, the workspace switches to the corresponding object in JavaClient and locks the object to prevent editing.

  These functions can also be invoked by using the **context menu** within the integrated preview:
  - Show object
  - Edit object

- **Click on a link in the integrated preview:** The preview switches to the referenced object in JavaClient. If the "Workspace follows Preview" option is enabled, the active workspace also switches to the referenced object (see Figure 4-21).
- **Ctrl + click on a link in the integrated preview:** The preview does not switch to the referenced object in JavaClient, the current preview is retained. If the "Workspace follows Preview" option is enabled, the active workspace switches to the corresponding object or to the input component in JavaClient and does *not* follow the link of the clicked object in the preview. For example, if the editor wants to open a link input component, with a single click they would merely follow the link within the preview. If instead, the input component is to be opened with the link, the editor can perform the keyboard shortcut Ctrl + click on the referenced object (see Figure 4-21).
- **Ctrl + Shift + click on a linked object in the integrated preview:** In addition to the response described above (see “Ctrl + click on a link“), the Shift key can be used to directly lock the object to prevent editing (see Figure 4-21).



Simple click within the integrated preview follows the referenced page.

Link to page "About us"

Ctrl + shift + click within the integrated preview opens and locks the input component for editing.

Ctrl + click within the integrated preview opens the input component.

About us
Mithras Energy has set itself the goal of innovatively developing and efficiently selling solar energy products.
About us
german
english
Mithras Energy

Figure 4-21: Click on a link within the integrated preview

The integrated preview is configured using the new "View" menu (cf. Chapter 8.1.7 page 117).

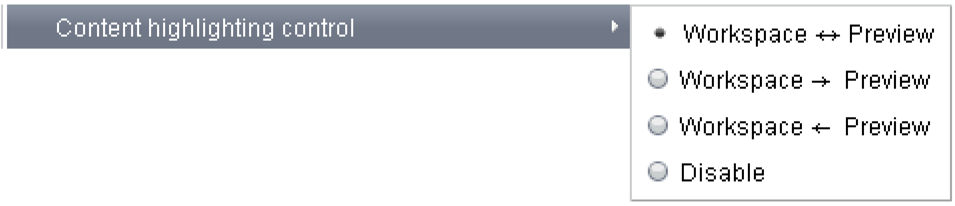


Figure 4-22: "View" menu – Configure control for Content Highlighting

- **Workspace follows preview (and vice-versa):** Click in the active workspace to display the matching element in the integrated (inline) preview and vice-versa.
- **Preview follows workspace:** Click in the active workspace to display the matching element in the integrated preview, but not vice-versa.
- **Workspace follows preview:** Click in the integrated (inline) preview to display the matching element in the active workspace, but not vice-versa.

In addition, it is also possible to enable "Content Highlighting" for an external web server (**"Workspace Follows External Web Browser"**). In this case, only one direction is supported: A click in the preview open in an external web browser displays the matching element in the active workspace, but not vice-versa.

Analogous to use of Easy-Edit, the templates of a project must be adjusted first for use of the “Content Highlighting" function (see Chapter 5.1.7 page 86).

Note: The “Content Highlighting“ functionality described in this chapter does only have an effect on the preview of the HTML output channel. Other output channels are not taken into account.

Note: FirstSpirit components do not all completely support the "Content Highlighting" function. This is especially true for multi-valued components, for example, CMS_INPUT_CONTENTAREALIST.

Note: Content Highlighting is not supported for display of the integrated preview in an external window (cf. Chapter 4.1.4 page 25).

The "Content Highlighting" function is purely a JavaClient function and is not supported in WebEdit (see Chapter 4.3 page 74).

4.1.6 Media gallery

From FirstSpirit Version 4.2, pictures from the Media Store can be output in a gallery view. The templates and database tables which are required for this must first be configured by the template developer for a project (see Chapter 5.1.4 page 81). According to the particular project they can differ from the description in this chapter.

Two forms of display are possible:

Single galleries: These mostly consist of a homepage in which general information about the gallery, such as its name and description, are displayed and one or several teaser pictures, which are usually gallery pictures displayed with reduced size. The gallery's name, description and teaser pictures are linked. To this end, a reference to the medium from the Media Store is rendered within the template for each picture in the gallery, mostly with a larger resolution. All the pictures of a gallery can be gathered together in a group by an attribute within the template. This enables the display of a picture navigation ("picture 2 of 3") within the enlarged gallery view and paging through all the pictures in a gallery.

Gallery overviews: If there are several galleries in a project, depending on the template developer's definition, these can be output filtered by categories (e.g. sport, politics, science) by defining queries.

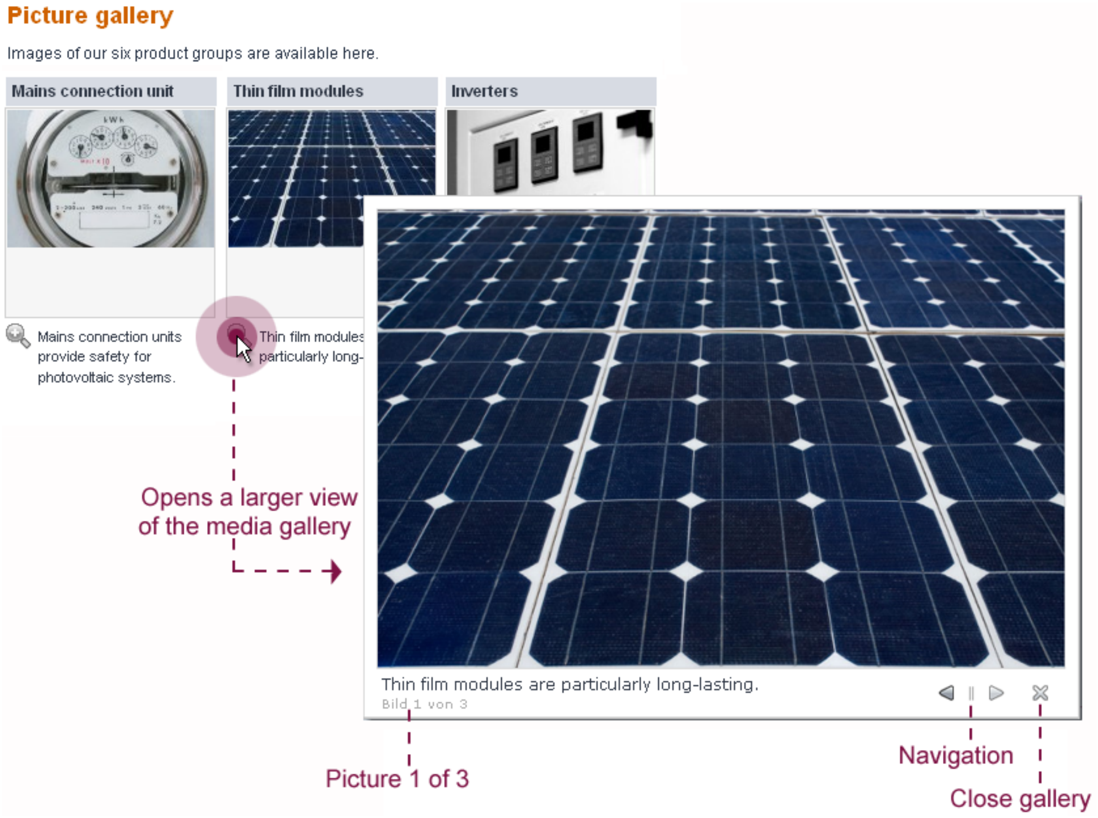
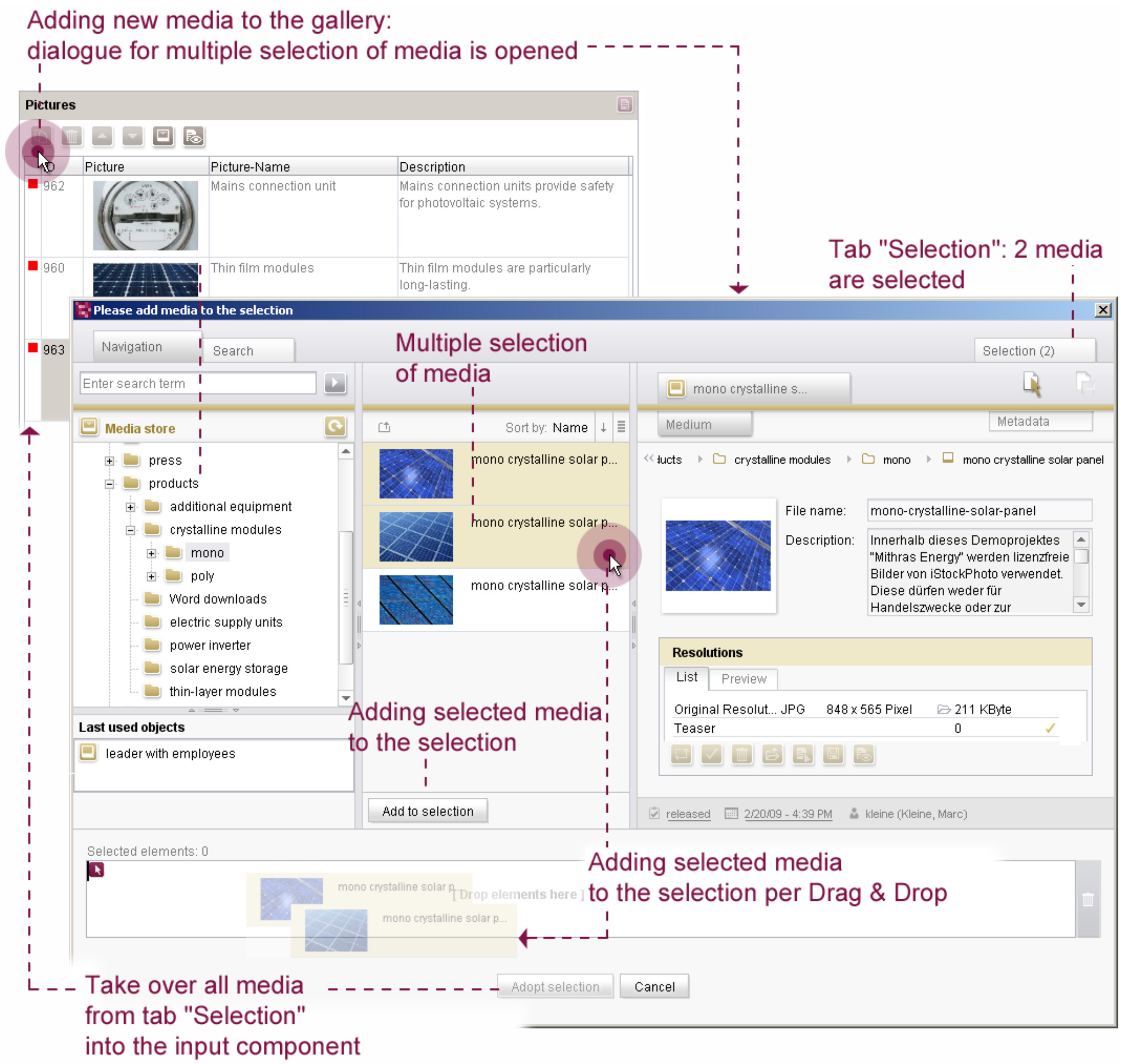

Picture gallery
Images of our six product groups are available here.
Mains connection unit
Thin film modules
Inverters
Mains connection units provide safety for photovoltaic systems.
Thin film modules particularly long-
Opens a larger view of the media gallery
Thin film modules are particularly long-lasting.
Bild 1 von 3
Picture 1 of 3
Navigation
Close gallery

Figure 4-23: Gallery entry page with open media galleries

The pictures to be used in the galleries must first (like all other media too) exist in the Media Store. All other data such as names and descriptive texts of the individual media is, on the other hand, stored and managed in data sources.

The enhanced input component CMS_INPUT_CONTENTLIST is available within the Content Store for maintenance of the galleries (see Chapter 4.1.6 page 35). Enhancement of the CMS_INPUT_CONTENTLIST input component enables the selection of several media from the project's Media Store and their arrangement and display within a gallery view.



Adding new media to the gallery: dialogue for multiple selection of media is opened

Tab "Selection": 2 media are selected

Multiple selection of media

Adding selected media to the selection

Adding selected media to the selection per Drag & Drop

Take over all media from tab "Selection" into the input component

File name: mono-crystalline-solar-panel

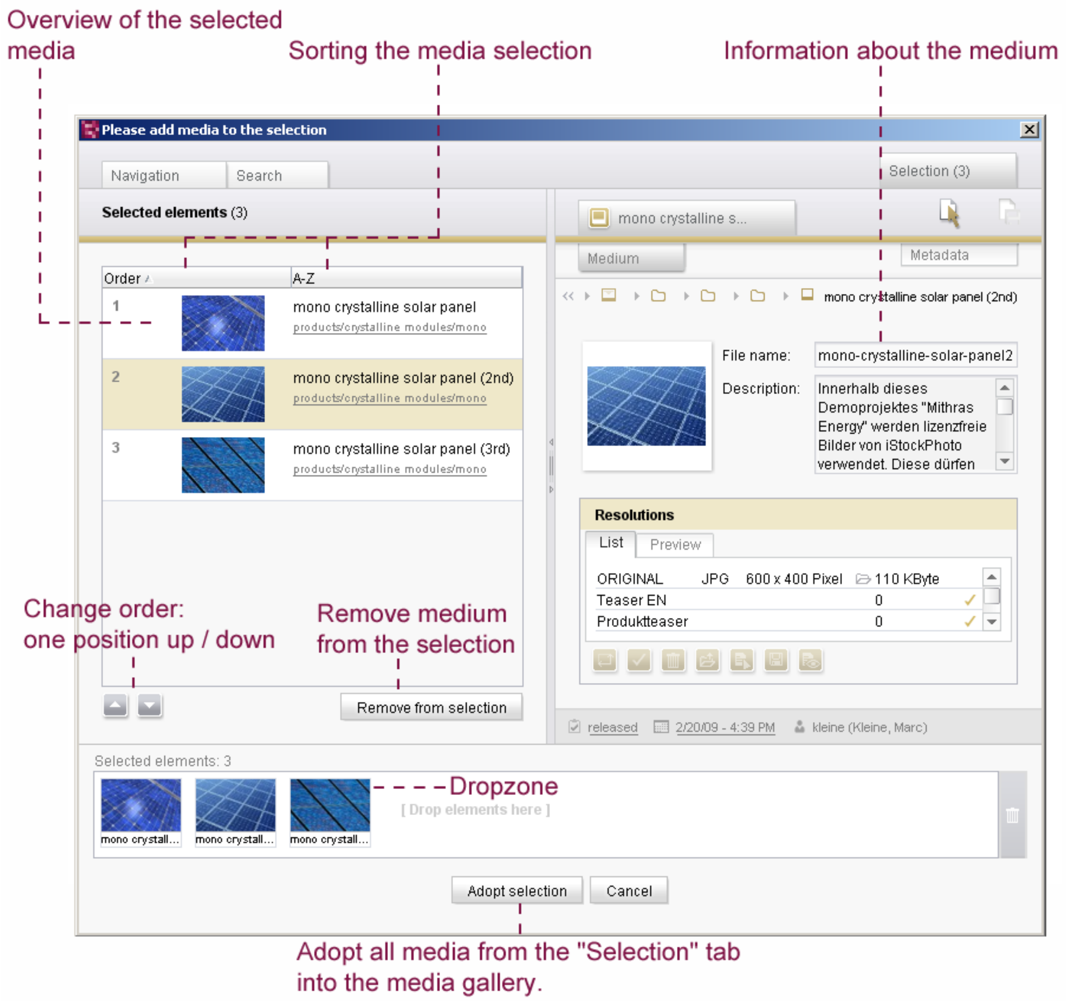
Description: Innerhalb dieses Demoprojektes "Mithras Energy" werden lizenzfreie Bilder von iStockPhoto verwendet. Diese dürfen weder für Handelszwecke oder zur

Original Resolut... JPG	848 x 565 Pixel	211 KByte
Teaser		0 ✓

Figure 4-24: Adding pictures to the media gallery (selection dialog)

Adding pictures using a selection dialog: When adding elements to the input component (media gallery), the new dialog for multiple selection of media opens (see Figure 4-24). Images only (not files) are displayed in the dialog. The dialog enables navigation within the Media Store (in the left-hand part of the dialog), single or multiple selection of media (using the middle part of the dialog) and the display of additional information on the selected medium (in the right-hand part of the dialog). The selected media can be easily added to a selection by means of Drag & Drop into the drop zone or by clicking the button.

The selection of all selected media can then be directly copied into the input component ("Adopt selection") or can be further edited first in the "Selection" tab (see Figure 4-25).



Overview of the selected media

Sorting the media selection

Information about the medium

Change order: one position up / down

Remove medium from the selection

Dropzone

Adopt all media from the "Selection" tab into the media gallery.

Order	A-Z
1	mono crystalline solar panel products/crystalline modules/mono
2	mono crystalline solar panel (2nd) products/crystalline modules/mono
3	mono crystalline solar panel (3rd) products/crystalline modules/mono

mono crystalline solar panel (2nd)

File name: mono-crystalline-solar-panel2

Description: Innerhalb dieses Demoprojektes "Mithras Energy" werden lizenzfreie Bilder von iStockPhoto verwendet. Diese dürfen

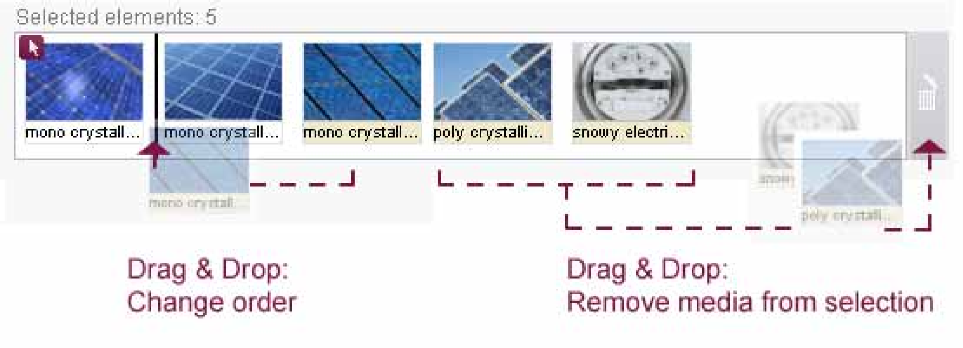
ORIGINAL	JPG	600 x 400 Pixel	110 KByte	
Teaser EN			0	✓
Produktteaser			0	✓

Figure 4-25: Tab selection: Edit selected media

The media selected to date can be sorted within the "Selection" tab. All media can be sorted in the list view, either by order (takes into account the time/date the media was added to the selection) or alphabetically (by reference or display name). Apart from this sorting option, the order of individual media can be changed in the overview.

Additional media can be added to the selection at any time by switching to the "Navigation" or "Search" tab. Media can be deleted using the "Remove from Selection" button (see Figure 4-25).

It is also possible to remove media from the selection (drop onto the Recycle Bin) and to sort individual elements by using Drag & Drop to place them in the drop zone:



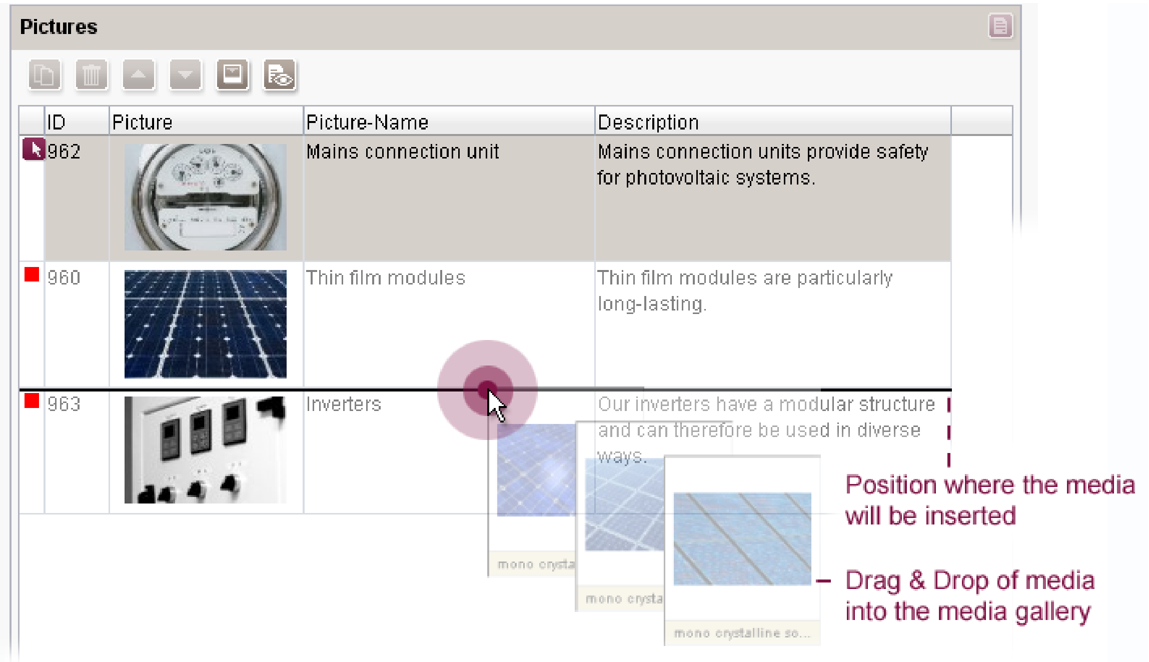
Drag & Drop:
Change order

Drag & Drop:
Remove media from selection

Figure 4-26: Sorting and removing media within the drop zone

When the selection is accepted, they are copied into the input component CMS_INPUT_CONTENTLIST in their sorted order.

Adding pictures by means of Drag & Drop: If the extensive functions of the media selection dialog are not required, media can also be simply added to the input component (media gallery) by means of Drag & Drop. In this case, the media (pictures only) can be selected first via the Media Store and then dragged by means of Drag & Drop into the input component in the Content Store (see Chapter 4.1.10 page 45).



	ID	Picture	Picture-Name	Description
	962		Mains connection unit	Mains connection units provide safety for photovoltaic systems.
🟥	960		Thin film modules	Thin film modules are particularly long-lasting.
🟥	963		Inverters	Our inverters have a modular structure and can therefore be used in diverse ways.

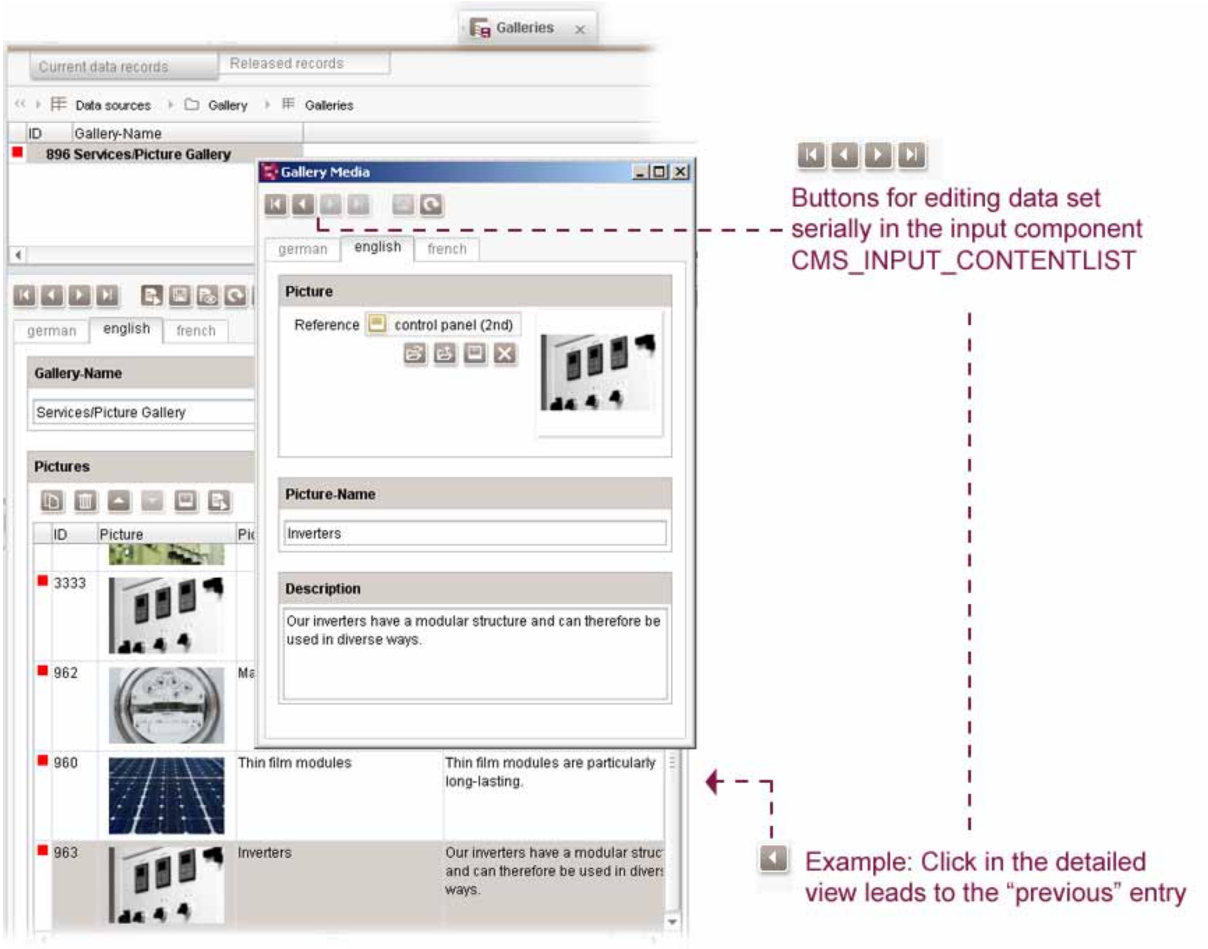
Position where the media will be inserted

Drag & Drop of media into the media gallery

Figure 4-27: Adding pictures to the media gallery (Drag & Drop)

In addition, the multiple selection option is available within the CMS_INPUT_CONTENTLIST input component. Several objects, for example media, can be selected from the overview. An action can then be performed on the multiple selection, for example, delete several media.

Serial editing in the input component CMS_INPUT_CONTENTLIST: A comfortable way to edit datasets in the input component serially has been developed within the scope of the enhancement of the media gallery. Using the icons ⏮ ◀ ▶ ⏭ in the upper area of the dialog with details the editor can switch directly to the first / previous / next / last dataset of the input component. If several consecutive entries are to be edited within the input component this dialog needs not to be closed after each change and reopened subsequently on another dataset. The editor can switch directly to the desired dataset and edit it in the detail dialog.



Gallery-Name: Services/Picture Gallery

Picture-Name: Inverters

Description: Our inverters have a modular structure and can therefore be used in diverse ways.

ID	Picture	Picture-Name	Description
960		Thin film modules	Thin film modules are particularly long-lasting.
963		Inverters	Our inverters have a modular struc and can therefore be used in divers ways.

Buttons for editing data set serially in the input component CMS_INPUT_CONTENTLIST

Example: Click in the detailed view leads to the "previous" entry

Figure 4-28: Serial editing within the input component Contentlist

The "media gallery" function is *__not__* *supported in WebEdit (see Chapter 4.3 page 74).*

4.1.7 Generic link editors

Editorial maintenance of links in FirstSpirit is carried out using screen forms which can be adjusted, project-specifically, by the template developer. The forms are based on link configurations which specify in detail, which input fields are available to the editor for the maintenance of links. From FirstSpirit Version 4.2, the configuration options are considerably enhanced by the introduction of generic link editors. The configuration of link templates can then, analogous to page and section templates, be created by adding input components into the form area. In this way, on creating a link configuration, template developers can fall back on all FirstSpirit input components and therefore adapt the input options of the editors far more project-specifically than before (see Chapter 5.1.2 page 77).

The conventional input options for links can of course also be generated using the new, generic editors. With this, differences in the display and user prompting of the input screen forms can result for the editors:

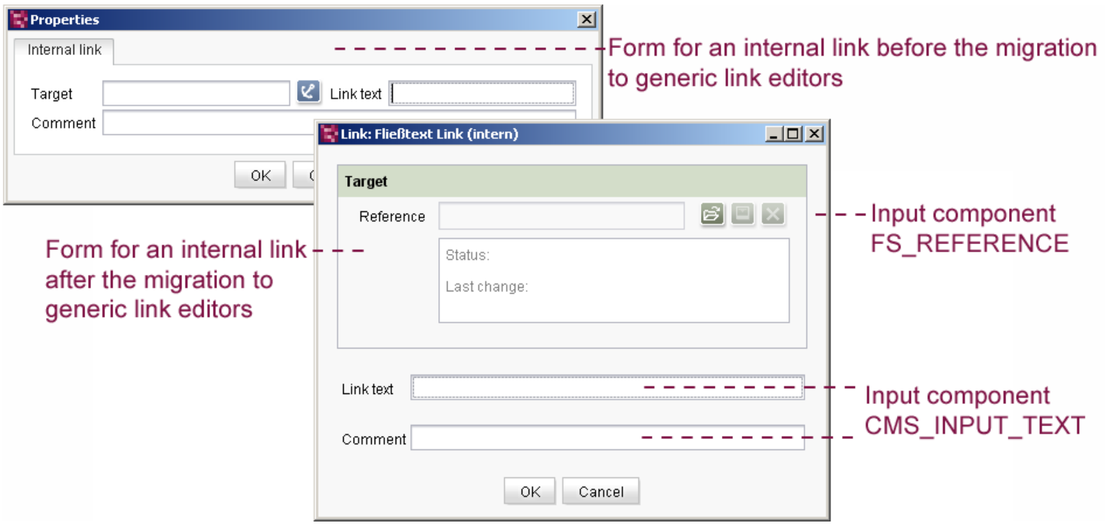

Properties
Internal link
Target
Link text
Comment
OK
Link: Fließtext Link (intern)
Target
Reference
Status:
Last change:
Link text
Comment
OK
Cancel
Form for an internal link before the migration to generic link editors
Form for an internal link after the migration to generic link editors
Input component FS_REFERENCE
Input component CMS_INPUT_TEXT

Figure 4-29: Form of an internal link before and after migration

Several new input components were introduced with FirstSpirit Version 4.2, to enable all the functions of the static link editors used to date to be reproduced on the new generic editors (see Chapter 5.1.9 page 88). For example, a new input component with a selection option for different reference types from several Stores has been created (FS_REFERENCE). The template developer can configure the selection dialog project specifically, for example, in order to reproduce the selection option of media references (pictures and files) used to date or to provided advanced selection options (pictures, files, page references, document groups, etc.) (see Figure 4-30).

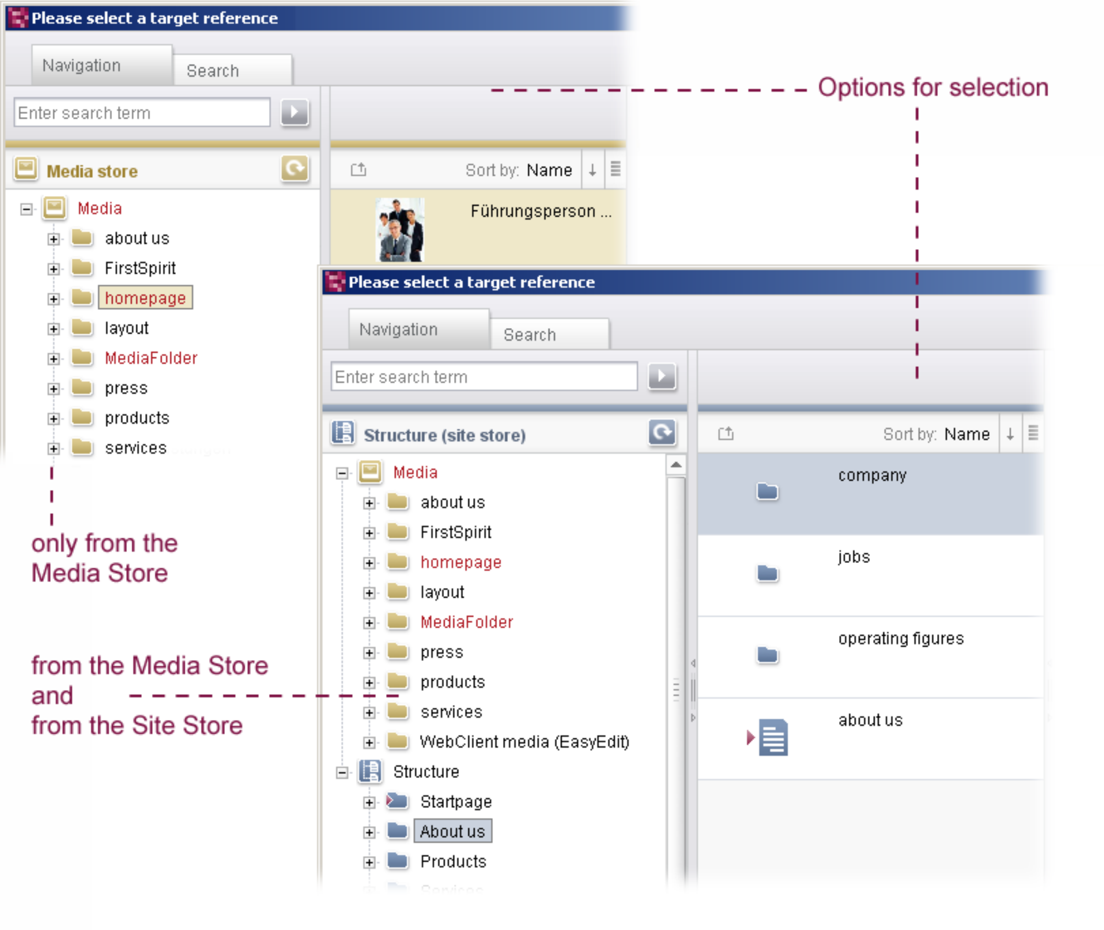
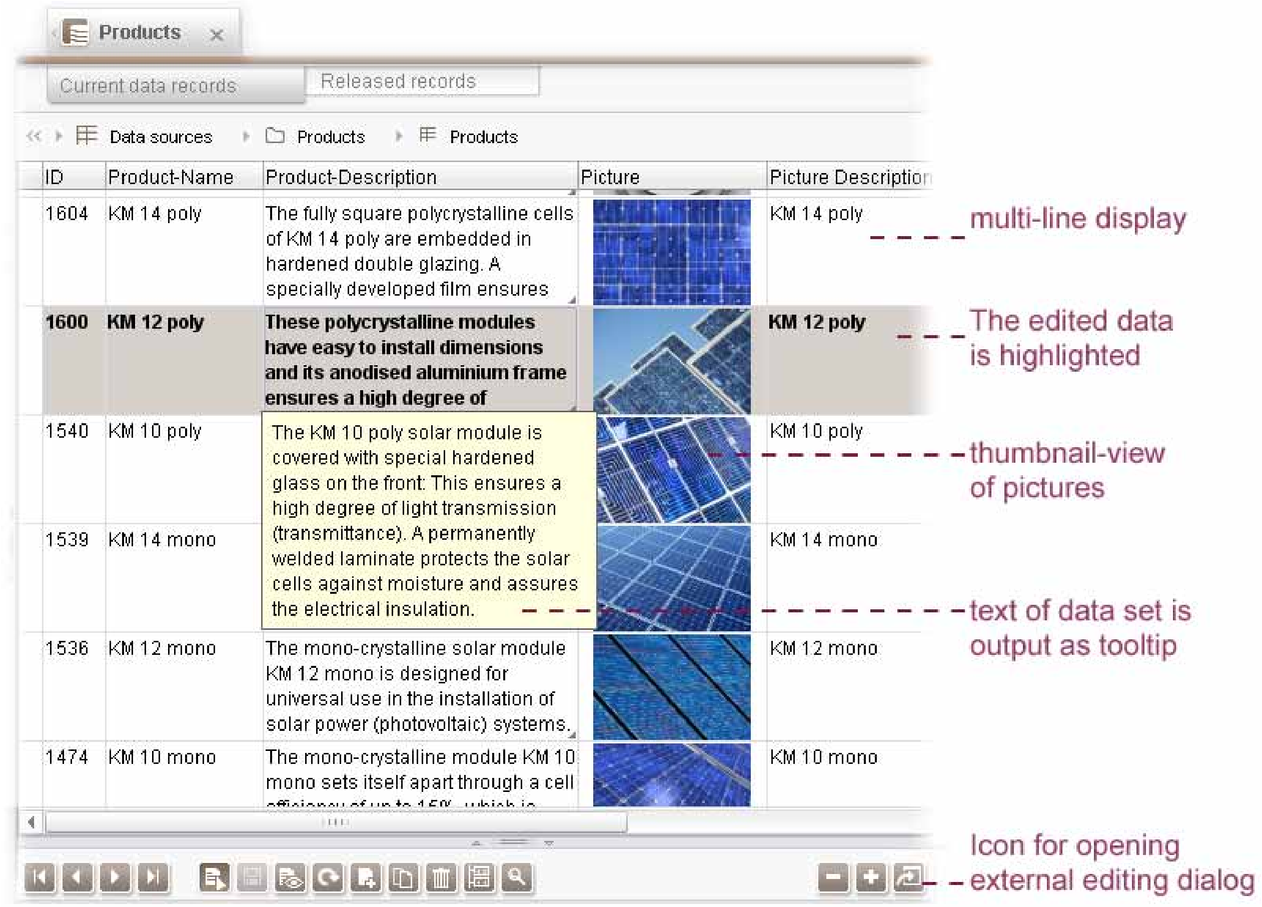


Figure 4-30: Selection dialog

The "Generic link editors" function is supported in WebEdit (see Chapter 4.3 page 74). However, there are restrictions to use of the new input components in WebEdit (see Chapter 4.2.3 page 71).

4.1.8 Optimised display of datasets

With FirstSpirit Version 4.2, the display of content within the Content Store is optimised for editorial work. To this end, options for configuring the views are made available within the Template Store. The new configuration options can be used to project-specifically adjust display of the views (see Chapter 5.1.3 page 79).



ID	Product-Name	Product-Description	Picture	Picture Description
1604	KM 14 poly	The fully square polycrystalline cells of KM 14 poly are embedded in hardened double glazing. A specially developed film ensures		KM 14 poly
1600	**KM 12 poly**	**These polycrystalline modules have easy to install dimensions and its anodised aluminium frame ensures a high degree of**		**KM 12 poly**
1540	KM 10 poly			KM 10 poly
1539	KM 14 mono			KM 14 mono
1536	KM 12 mono	The mono-crystalline solar module KM 12 mono is designed for universal use in the installation of solar power (photovoltaic) systems.		KM 12 mono
1474	KM 10 mono	The mono-crystalline module KM 10 mono sets itself apart through a cell		KM 10 mono

The KM 10 poly solar module is covered with special hardened glass on the front: This ensures a high degree of light transmission (transmittance). A permanently welded laminate protects the solar cells against moisture and assures the electrical insulation.

multi-line display

The edited data is highlighted

thumbnail-view of pictures

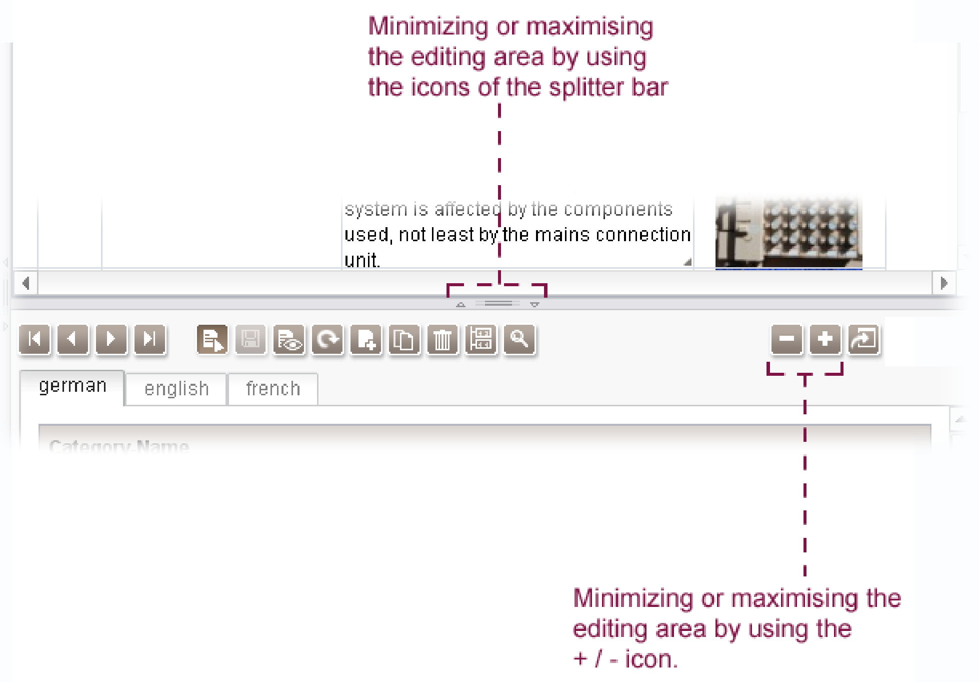
text of data set is output as tooltip

Icon for opening external editing dialog

Figure 4-31: Multi-line display of datasets

The project's template developer can specify the line height of the entries and the column width, as well as the order of the columns. If the text to be displayed within an input component is larger than the area for the component available in the overview, the text is inserted as a tooltip when the mouse cursor is passed over it. Pictures within a multi-line display are now inserted in the overview as thumbnails. In addition, display of the content of the CMS_INPUT_CONTENTAREALIST input component is now supported and this is also displayed in the overview.

The view within the Content Store can now also be minimised or maximised using the splitter bar (between the overview and editing area) instead of the +/- buttons (see Figure 4-32). Moreover, the editing area can be displayed automatically in the bottom area of the JavaClient by double clicking a dataset in the list view. The dataset can be edited in an external dialog by clicking the icon “Open big editing window“ (see Figure 4-31).



Minimizing or maximising the editing area by using the icons of the splitter bar
system is affected by the components used, not least by the mains connection unit.
german
english
french
Category Name
Minimizing or maximising the editing area by using the + / - icon.

Figure 4-32: Maximising and minimising the editing area

When editing the currently edited dataset is now more highlighted and displayed in bold within the list of datasets (see Figure 4-31). So the editor can identify at a single glance to the list the dataset which is edited at the moment. If the editor selects a new dataset in the list the focus will change in the bottom editing area, and the now edited dataset will be highlighted in bold.

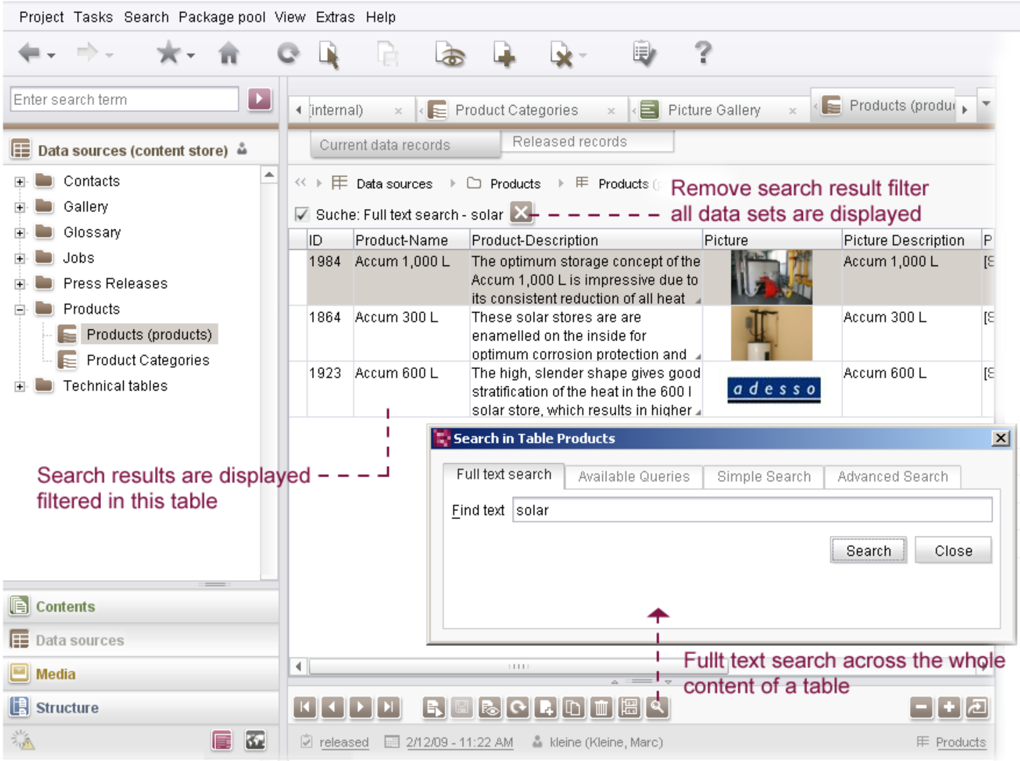
The "multi-line display of datasets" is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.9 Full text search in the Content Store

From FirstSpirit Version 4.2, a full text search is available in the Content Store's table view.

Until now, the data search could merely be used to query individual text columns in the "Simple Search". The new full text search now enables searching in all the text content of a table, including the texts deposited in input components of the type DOM Editor.

The search dialog is opened by clicking the 🔍 "data search" icon. A search term can be entered in the "Full Text Search" tab. All the contents of the table are then searched for this term. If several search terms are entered, the search for these terms is based on an AND operation. The search is case sensitive, i.e. takes into account upper and lower case letters in the search terms. Foreign key relations with other tables are not taken into account.



☑ Suche: Full text search - solar

ID	Product-Name	Product-Description	Picture	Picture Description	P
1984	Accum 1,000 L	The optimum storage concept of the Accum 1,000 L is impressive due to its consistent reduction of all heat		Accum 1,000 L	[S
1864	Accum 300 L	These solar stores are are enamelled on the inside for optimum corrosion protection and		Accum 300 L	[S
1923	Accum 600 L	The high, slender shape gives good stratification of the heat in the 600 l solar store, which results in higher	adesso	Accum 600 L	[S

Find text: solar

Remove search result filter all data sets are displayed

Search results are displayed filtered in this table

Fullt text search across the whole content of a table

Figure 4-33: Full text search in the Content Store

The table view then shows a filtered view of the search results. The filtered view can be cancelled by clicking the ☒ icon.

"Full Text Search in the Content Store" is *__not__* *supported in WebEdit (see Chapter 4.3 page 74).*

4.1.10 Drag & Drop FirstSpirit objects

Several new "Drag & Drop" functions have also been implemented within the scope of FirstSpirit Version 4.2.

Drag & Drop from the local file system into JavaClient: Example: Media from the workstation computer's local file system can be dragged directly into the Media Store of the FirstSpirit project by means of Drag and Drop. Both individual and multiple selection is supported. If the media within the file system is structured in folders, the folders can also be optionally generated in the Media Store with the Drag & Drop. This functional already exists from FirstSpirit Version 4.1, but is mentioned again here for reasons of completeness.

Drag & Drop from JavaClient into the local file system: Any objects (e.g. media, pages, folders, templates) from the FirstSpirit project can be dragged directly into the local file system of the workstation computer by means of Drag & Drop. Folders, pages and other FirstSpirit objects are created as Zip files and media in the respective file format in the local file system (analogous to the conventional context menu function "Export/Import").

Both individual and multiple selection are supported for Drag & Drop. The objects can either be selected from the tree or directly from the thumbnail overview in the middle workspace.

If an individual or multiple selection is made from the tree view or thumbnail view, the drag action shows a preview of the element to be moved (see Figure 4-34).

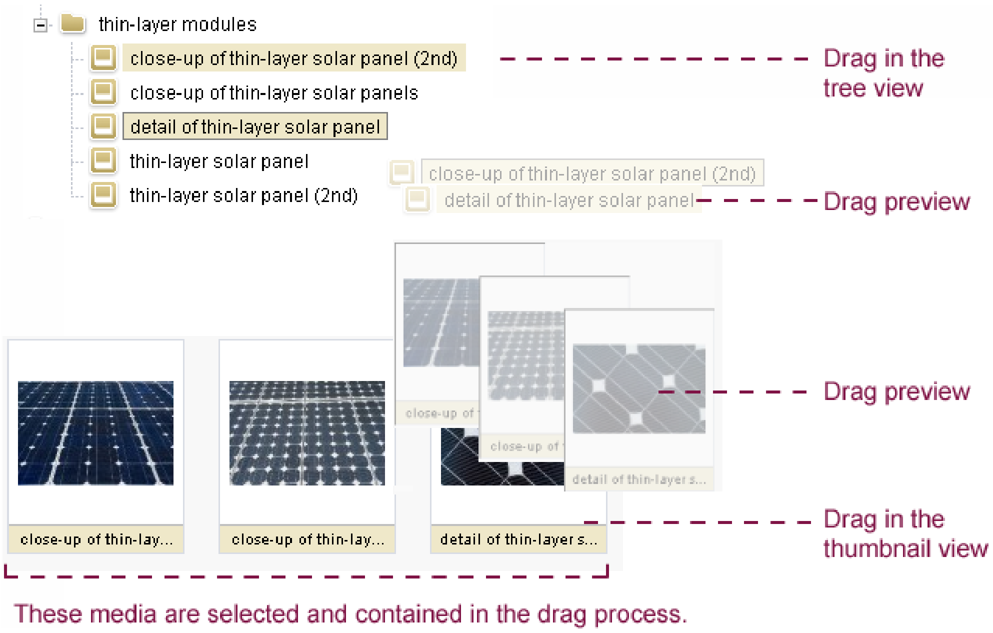
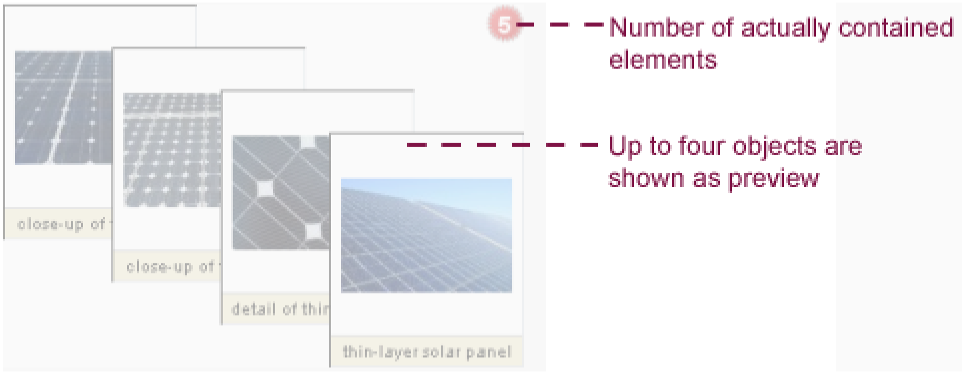

thin-layer modules
close-up of thin-layer solar panel (2nd)
close-up of thin-layer solar panels
detail of thin-layer solar panel
thin-layer solar panel
thin-layer solar panel (2nd)
close-up of thin-layer solar panel (2nd)
detail of thin-layer solar panel
Drag in the tree view
Drag preview
Drag preview
Drag in the thumbnail view
close-up of thin-lay...
close-up of thin-lay...
detail of thin-layer s...
These media are selected and contained in the drag process.

Figure 4-34: Drag action in the tree and in the thumbnail overview

The "drag" preview shows no more than four elements. If the selection contains more than four elements, the number of elements is also shown (see Figure 4-35):



5
Number of actually contained elements
Up to four objects are shown as preview
close-up of
close-up of
detail of thir
thin-layer solar panel

Figure 4-35: Drag action with more than four elements

Drag & Drop within the thumbnail view (media): The media (and media folders) within the Media Store can also be moved (and copied) using the thumbnail view.

Both individual and multiple selection are supported for Drag & Drop. A multiple selection can be made in the thumbnail view using Ctrl + A (select all objects in the overview), using Shift + click (select area within the overview) or using Ctrl + click (select objects one after the other)

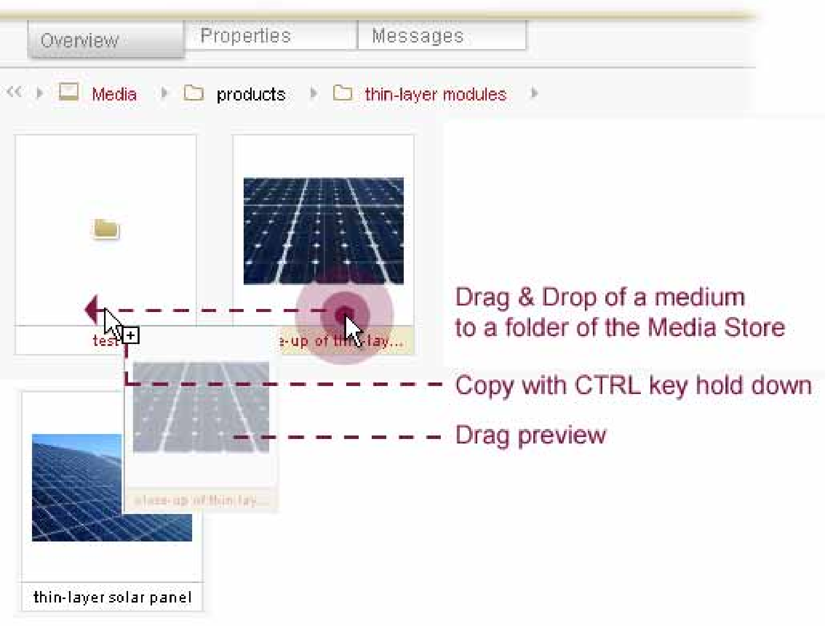


Figure 4-36: Drag & Drop media in the thumbnail view

The required media (or folders) can be selected and moved or copied into other media folders within the thumbnail view. The Ctrl key must be kept pressed during the "Drag & Drop" action to copy media into a folder. The copying of media (or folders) is visualised by a ⊞ on the mouse cursor. Simple Drag & Drop is sufficient to move a medium into a folder.

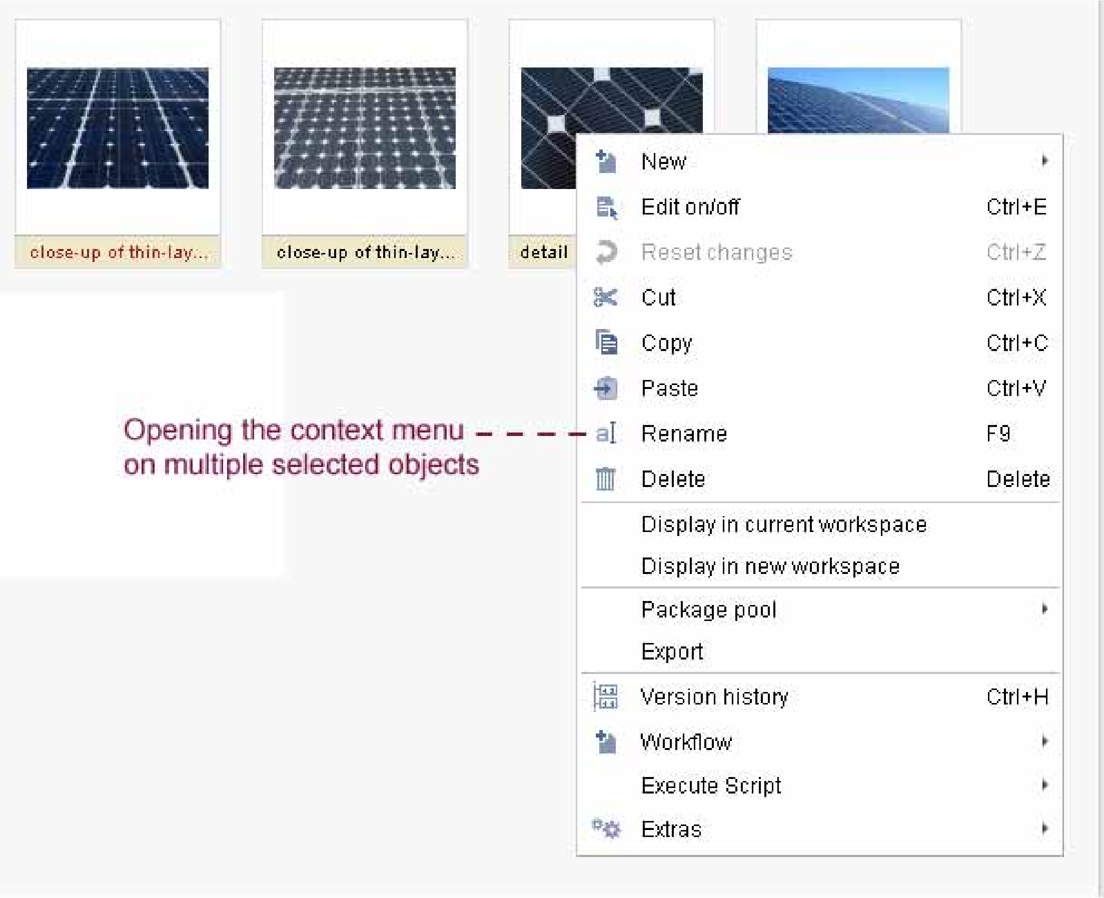


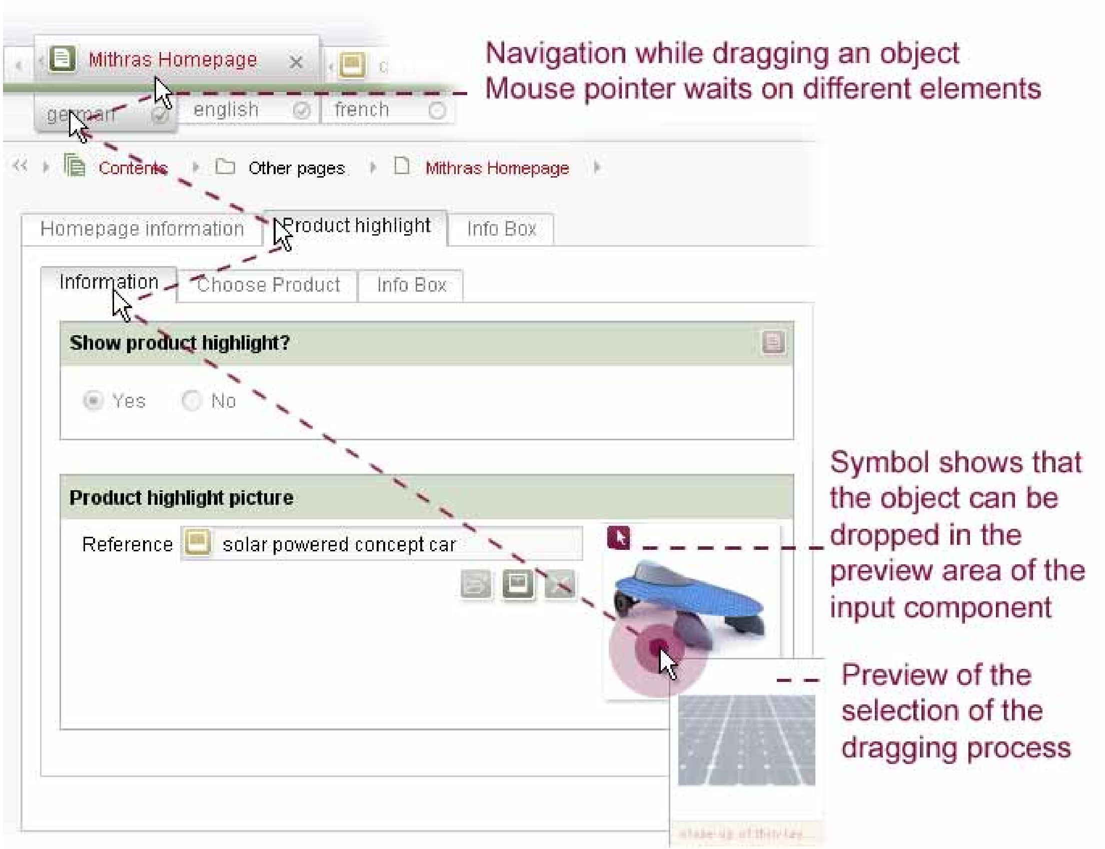
Figure 4-37: Opening the context menu on a multiple selection

All functions can also be performed using the context menu, within the tree view or the thumbnail view, on a single selection or a multiple selection.

Drag & Drop between two workspaces: Apart from editing in individual workspaces, it is also possible to Drag & Drop between the workspaces. For example, media (or other FirstSpirit objects) from the thumbnail view of a workspace can be copied into the input component of another workspace by means of Drag & Drop (see Chapter 4.1.1 page 17).

For example, the media can be selected by clicking the preview image in the Media view and, keeping the left-hand mouse button pressed, can be dragged onto the required workspace. The medium can then be dropped into the preview area of the corresponding input component. A drag option is indicated in JavaClient by the icon.

Navigation between the required areas is performed using the mouse cursor during the "Drag" action. If the mouse cursor lingers over the required area, for example an inactive workspace, it is opened. This type of navigation is possible on the tabs of the open workspaces and the sub-tabs within the workspaces (languages or output channels). In addition, the navigation can also be performed on the tabs of grouping elements of a form (see Figure 4-38).



Reference: solar powered concept car

Navigation while dragging an object
Mouse pointer waits on different elements

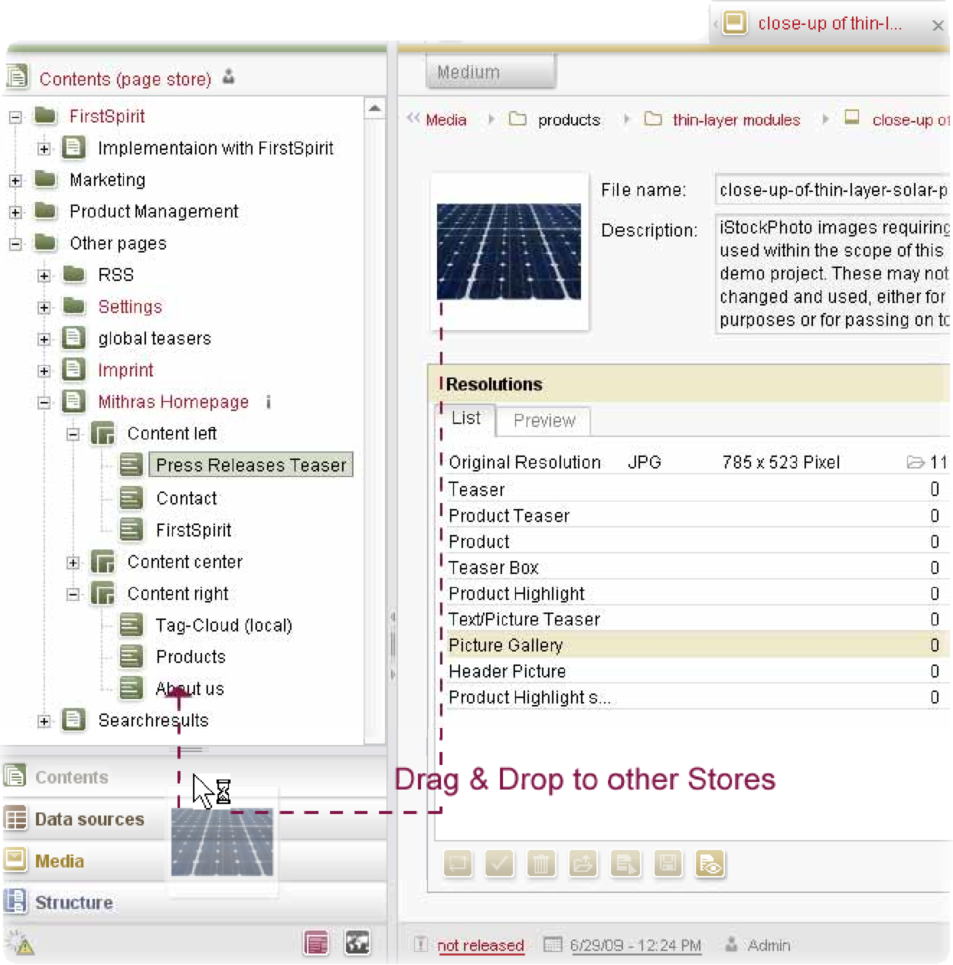
Symbol shows that the object can be dropped in the preview area of the input component

Preview of the selection of the dragging process

Figure 4-38: Navigation during a drag action

It is also possible to scroll over the workspaces' tabs if the mouse cursor lingers over the scroll symbols ◂ ▸. Automatic scrolling within the form area (and within the tree view of course) is enabled during a drag action, as soon as the mouse cursor approaches the boundaries of the respective area. For example, to go to the bottom end of the form, simply move the mouse cursor towards the bottom edge of the workspace.

Inter-Store Drag & Drop: It is also possible to Drag & Drop across two Stores, by selecting the element, for example a medium, within the workspace or tree view of a Store first and then, pressing the left-hand mouse button on the button of the required Store, the element is dragged into the left-hand navigation area of JavaClient. The store level is automatically expanded.



File name: close-up-of-thin-layer-solar-p

Description: iStockPhoto images requiring used within the scope of this demo project. These may not changed and used, either for purposes or for passing on to

Resolutions

Original Resolution	JPG	785 x 523 Pixel	11
Teaser			0
Product Teaser			0
Product			0
Teaser Box			0
Product Highlight			0
Text/Picture Teaser			0
Picture Gallery			0
Header Picture			0
Product Highlight s...			0

Figure 4-39: Drag & Drop across stores

Further navigation within the new Store can be performed using the tree view. If the mouse cursor lingers over the required element in the tree, for example, a page or a section, this element is opened in the active workspace. Navigation in the active workspace takes place as described under "Drag & Drop between two workspaces".

Drag & Drop from the search dialog into JavaClient: The new (non-modal) search dialog for the quick text search (see Chapter 4.1.15 page 58) also supports Drag & Drop of the search results into JavaClient. The FirstSpirit objects found can be dragged directly from the search dialog into the tree view of JavaClient, the thumbnail view for media, the workspace or an input component. In this way, for example, the editor can drag a medium directly from the search dialog into the picture input component of a workspace and drop it there (see Figure 4-47).

Drop onto input components: In FirstSpirit Version 4.2, Drag & Drop is supported for the input components CMS_INPUT_PICTURE, CMS_INPUT_FILE, CMS_INPUT_CONTENTLIST and for the new input component FS_REFERENCE. The procedure on switching workspaces has already been described under "Drag & Drop between two workspaces". The object can then be dropped into the preview area of the picture input component or into the reference field. (Only the reference field is available as a "drop" option in the CMS_INPUT_FILE input component):

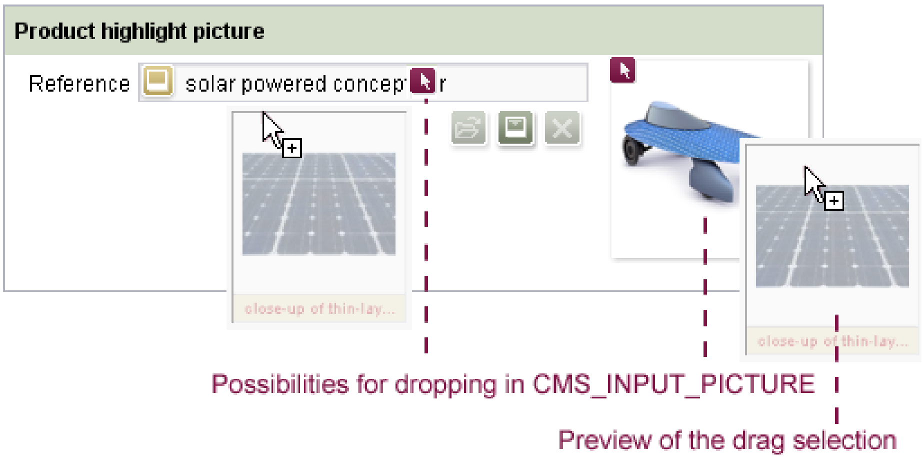

Product highlight picture
Reference
solar powered concep r
close-up of thin-lay...
close-up of thin-lay...
Possibilities for dropping in CMS_INPUT_PICTURE
Preview of the drag selection

Figure 4-40: Drop into a picture input component

If restrictions to certain folders in the Media Store are defined within the template in the input component, only media from these folders may be referenced in the input component. If a medium is dropped from a folder other than those which are allowed, an error message appears and the medium is not referenced within the input component.

Within the scope of enhancement of the CMS_INPUT_ CONTENTLIST input component for the "media gallery" function, Drag & Drop has also been realised for media in these input components. However, for this, the `MEDIAMODE` parameter must be enabled first for the input component. Unlike the picture input component, the CMS_INPUT_CONTENTLIST also takes into account the position of the medium. The current position is visualised during the "drop" action by a black line (see Figure 4-27).

Drop onto elements which are not yet open in a workspace: If, during the "drag" action, the editor notices that the required page (or another element) is not yet open in a workspace, the navigation can also take place via the tree view. Analogous to navigation between workspaces during a "drag" action (see Figure 4-38); the navigation is also carried out via the tree view.

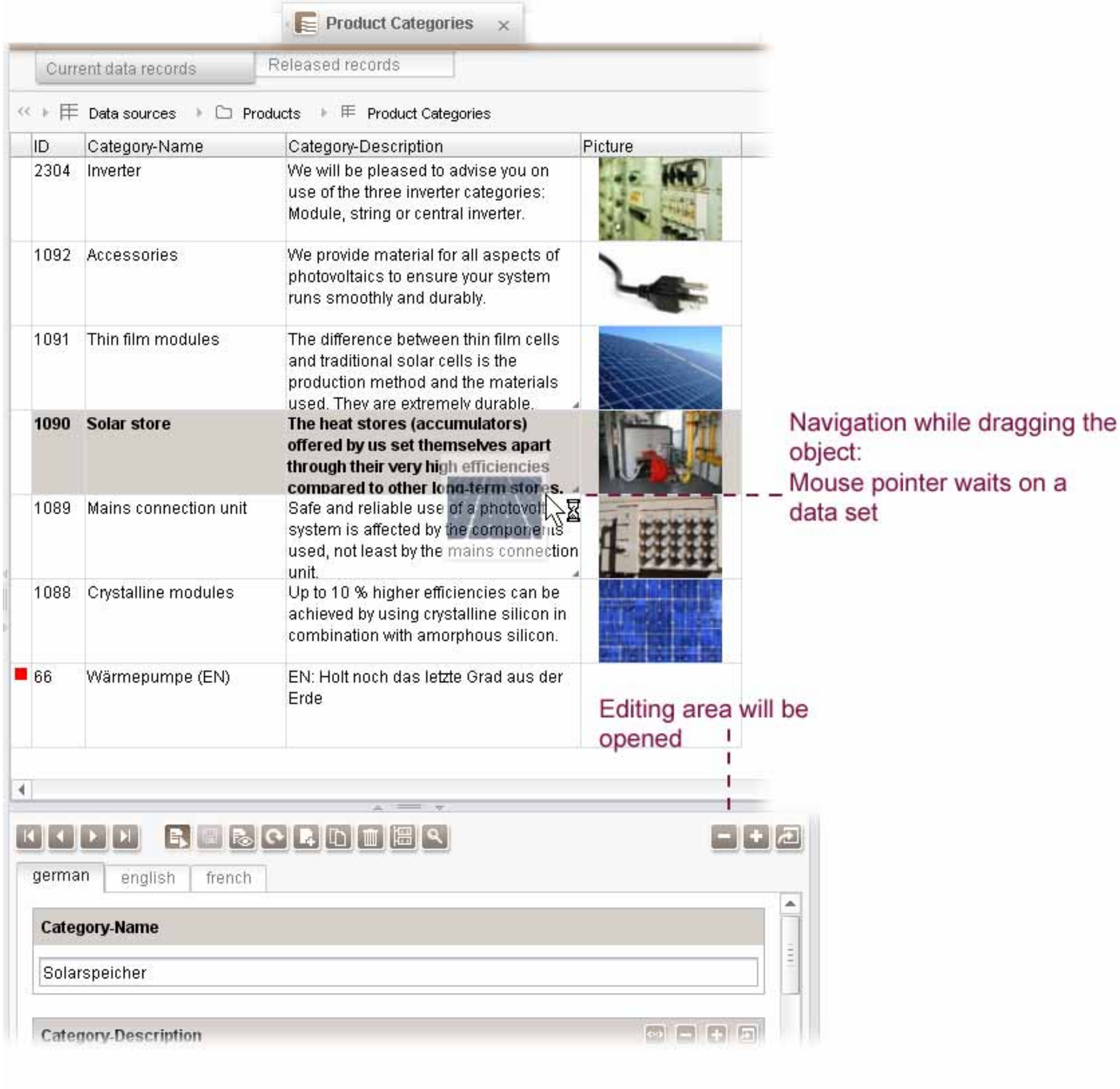
If the mouse cursor lingers:

- Over the ⊞ icon, the node in the tree is expanded.
- Over the icon or lettering of the element in the tree, for example, a page or a section, this element is opened in the active workspace.

Navigation in the active workspace takes place as described under "Drag & Drop between two workspaces".

Drop onto elements which are not in edit mode: If, during the "drag" action, the editor notices that the required page (or another element) has not yet been locked to prevent editing, they can also subsequently switch to edit mode. If the object, for example a medium, is dropped in a drop option, for example, dropped into the reference field of a picture input component, a message appear which gives the editor the opportunity to switch to edit mode. If the prompt is confirmed with "Yes", the object is then locked to prevent editing and the object is referenced in the input component. The "Lock-on-Drop" function is supported for the input components: CMS_INPUT_ PICTURE, CMS_INPUT_FILE, CMS_INPUT_CONTENTLIST and FS_REFERENCE.

Drag & Drop via the Content Store overview: When using Drag & Drop on FirstSpirit objects in the Content Store, the required editing area can be opened via the overview of the datasets. Navigation within the overview is performed using the mouse cursor during the "Drag" action. If the mouse cursor lingers over the required dataset, the dataset's editing area automatically opens for adding the object into an input component Another navigation option is to use the maximise and minimise buttons of the editing area. In this case, the editing area of the dataset currently selected in the overview is opened if the mouse cursor lingers over the +- button.



ID	Category-Name	Category-Description	Picture
2304	Inverter	We will be pleased to advise you on use of the three inverter categories: Module, string or central inverter.	
1092	Accessories	We provide material for all aspects of photovoltaics to ensure your system runs smoothly and durably.	
1091	Thin film modules	The difference between thin film cells and traditional solar cells is the production method and the materials used. They are extremely durable.	
1090	**Solar store**	**The heat stores (accumulators) offered by us set themselves apart through their very high efficiencies compared to other long-term stores.**	
1089	Mains connection unit	Safe and reliable use of a photovoltaic system is affected by the components used, not least by the mains connection unit.	
1088	Crystalline modules	Up to 10 % higher efficiencies can be achieved by using crystalline silicon in combination with amorphous silicon.	
🟥 66	Wärmepumpe (EN)	EN: Holt noch das letzte Grad aus der Erde	

Category-Name: Solarspeicher

Category-Description

Navigation while dragging the object:
Mouse pointer waits on a data set

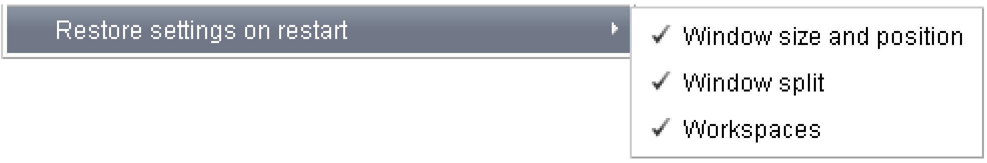
Editing area will be opened

Figure 4-41: Drag & Drop via a Content Store overview

The "Drag & Drop FirstSpirit Objects" function is _not_ *supported in WebEdit (see Chapter 4.3 page 74).*

4.1.11 Restore settings on restart

The new "View" menu in FirstSpirit JavaClient (cf. Chapter 8.1.7 page 117) can now be used not only to restore the window size and position but also to restore the positions of the separator between the tree and workspace and between the workspace and preview on restarting the JavaClient.



Restore settings on restart

✓ Window size and position

✓ Window split

✓ Workspaces

Figure 4-42: Menu view – Restore settings on restart

Window size and position: The window size and position set when JavaClient is exited, is adopted and is automatically restored with the next restart.

Window layout: The window layout of the navigation area, workspaces and (if set) the preview, set when JavaClient is exited, is adopted and is automatically restored with the next restart.

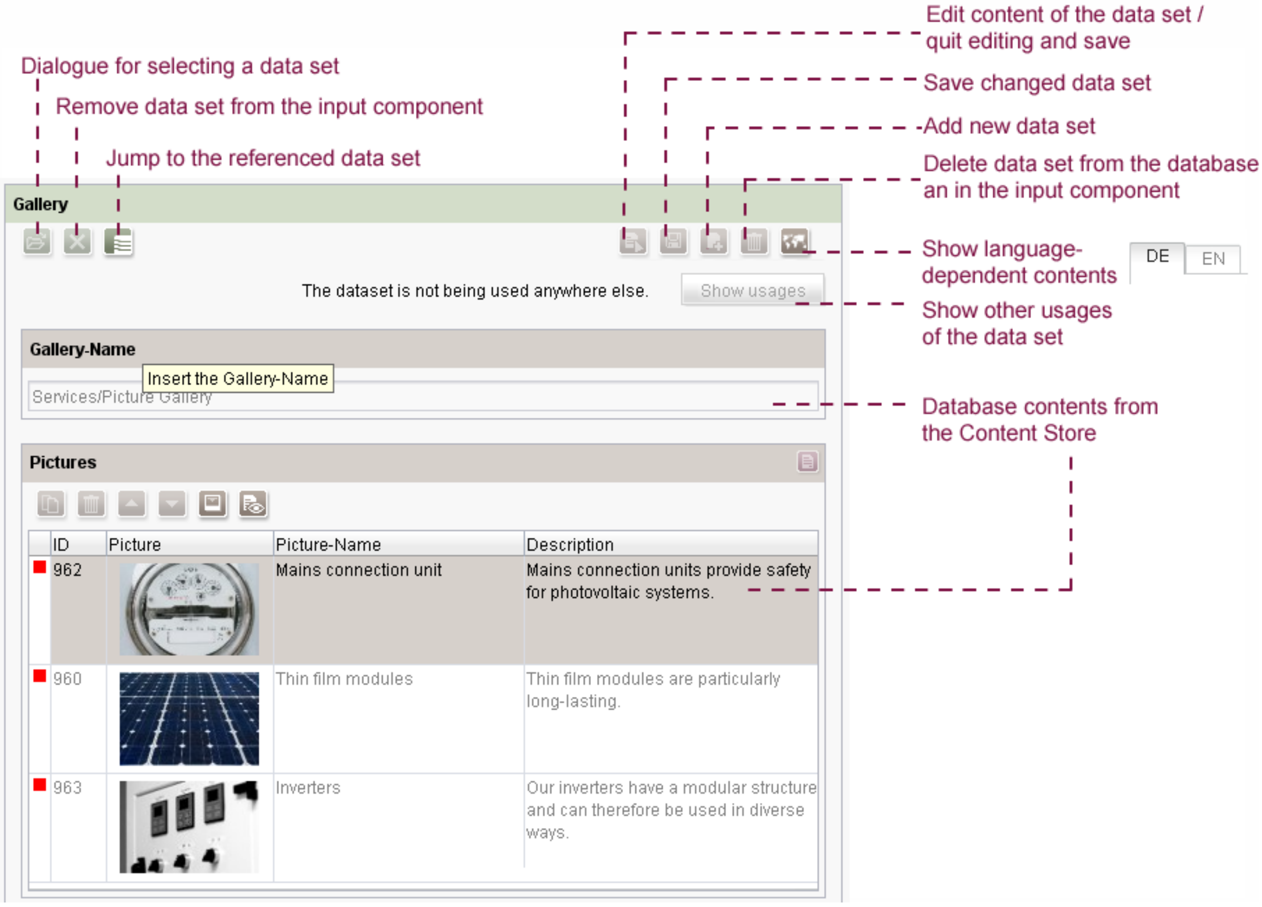
Workspaces: The configuration of the currently open workspaces can be saved at defined times ("Save Open Workspaces") or on exiting JavaClient ("Save on Exiting") (see Figure 4-4). These workspaces can be automatically reopened the next time JavaClient is started.

The "Restore Settings on Restart" function is purely a JavaClient function and is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.12 CMS_INPUT_OBJECTCHOOSER enhancement

The CMS_INPUT_OBJECTCHOOSER input component can be used to reference a dataset from the Content Store. The choice of target table is predefined by the template developer using the `table` parameter. In the selection dialog, the editor can therefore only select the required dataset, but not the target table (cf. Figure 4-44).

Analogous to the media objects of the media gallery, the referenced dataset here is also only a reference to the respective object in the Content Store. But the input component also provides direct editing options for the dataset. Coloured visualisation is predefined within the component to clearly demarcate with which fields and buttons the content of the dataset is edited and where, instead, only the reference of the dataset in the input component is concerned.


Dialogue for selecting a data set
Remove data set from the input component
Jump to the referenced data set
Edit content of the data set / quit editing and save
Save changed data set
-Add new data set
Delete data set from the database an in the input component
Show language-dependent contents
DE
EN
Show other usages of the data set
Database contents from the Content Store
Gallery
Show usages
Gallery-Name
Insert the Gallery-Name
Pictures

The dataset is not being used anywhere else.

Gallery-Name: Services/Picture Gallery

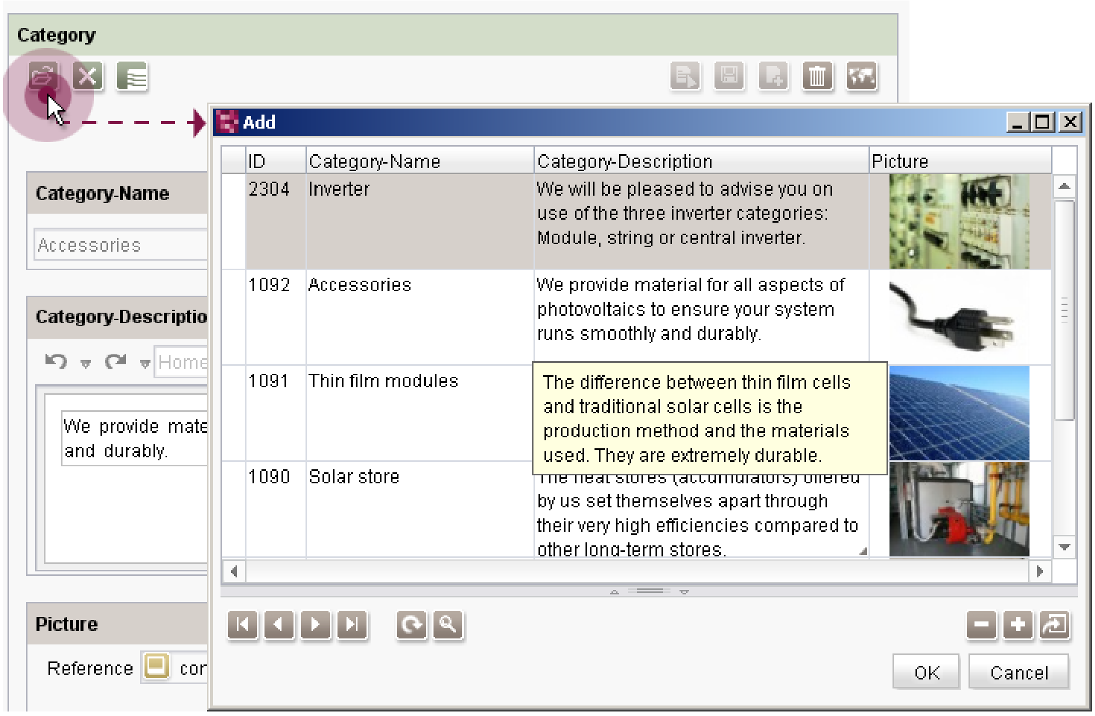
ID	Picture	Picture-Name	Description
962		Mains connection unit	Mains connection units provide safety for photovoltaic systems.
960		Thin film modules	Thin film modules are particularly long-lasting.
963		Inverters	Our inverters have a modular structure and can therefore be used in diverse ways.

Figure 4-43: Enhancements to the object chooser input component

The elements of the input component are displayed with the colour concept of the respective Store, in Figure 4-43 green for the Page Store. The elements which visualise content or editing options within the Content Store, on the other hand are displayed in the colour of the Content Store.

In addition, further new functions are implemented with FirstSpirit Version 4.2, in order to make working with this input component as convenient as possible:

- Ability to switch to the Content Store with the click of a button, in order to edit the selected dataset ("Jump to ...").
- The component not only enables the selected dataset to be removed from the input component, but also - depending on the user's permissions - to also delete the dataset in the Content Store.
- Show and hide language-dependent database content via tabs within the input component.
- If the dataset is referenced elsewhere, the number of uses is visualised in the input component. Click the button to display the other uses.



ID	Category-Name	Category-Description	Picture
2304	Inverter	We will be pleased to advise you on use of the three inverter categories: Module, string or central inverter.	
1092	Accessories	We provide material for all aspects of photovoltaics to ensure your system runs smoothly and durably.	
1091	Thin film modules	The difference between thin film cells and traditional solar cells is the production method and the materials used. They are extremely durable.	
1090	Solar store	The heat stores (accumulators) offered by us set themselves apart through their very high efficiencies compared to other long-term stores.	

Figure 4-44: Object chooser input component with selection dialog

The selection dialog for database contents can then be used to select the dataset (cf. Figure 4-44).

The enhancements of the CMS_INPUT_OBJECTCHOOSER input component are supported in WebEdit (see Chapter 4.2.4 page 71).

4.1.13 Extension of the reference graph for format templates

Essential functions of FirstSpirit are based on the so-called reference graph of a project. This had to be calculated for a project for the first time ever in Version 3.1 and since then has been constantly extended and developed. The reference graph is used to identify the dependencies of objects within complex projects.

Format templates used within the input components CMS_INPUT_DOM and CMS_INPUT_ DOMTABLE can now also be displayed using the reference graphs. When the context menu "Extras – Show Dependencies" is opened on a relevant section, the referenced format templates are displayed:

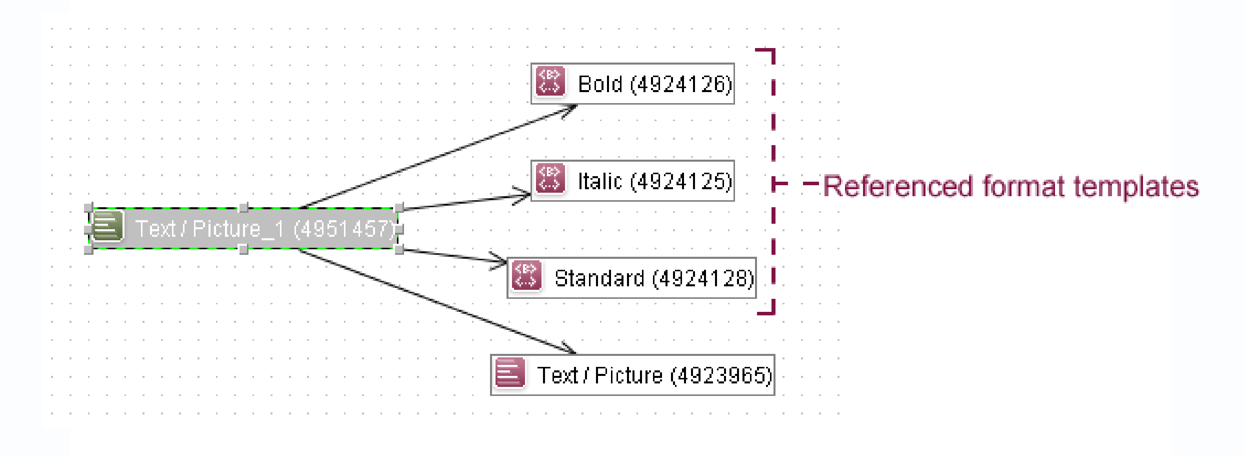


Figure 4-45: Reference graph with referenced format templates

This enhancement requires a change to the FirstSpirit Access API (see Chapter 8.1.8 page 117).

The "Reference Graph Extension" function is purely a JavaClient function and is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.14 Status display in the JavaClient's workspace

From FirstSpirit Version 4.2, a status indicator for the respective object is displayed with the following information in the active workspace of the FirstSpirit JavaClient:

- **Status** (released | not released | in workflow): Click on the entry to open the context menu for starting or set the workflow to the next step on the corresponding object.
- **Date of the last editing:** Click on this entry to open the version history of the object.
- **Last modifier**
- **Template:** Click on this entry to set the focus in the tree and in the workspace to the template of the object ("View template"). This function is only available if the editor has got the relevant permissions for the Template Store.

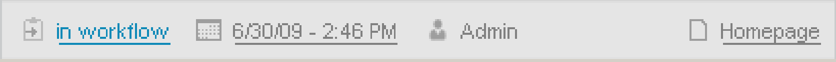


Figure 4-46: Status display for an element in the active workspace

The "JavaClient Status Display" function is purely a JavaClient function and is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.15 New quick text search in JavaClient (non-modal)

The search dialog for quick text search has been considerably enhanced for FirstSpirit Version 4.2. The new search dialog can be opened using the input field already familiar from Version 4.1, in the top left-hand area of JavaClient.

The search results are then displayed in the "Search" tab (see Figure 4-47). The left-hand area shows the Stores and the individual folders with the number of search results found, e.g. ⊞ ☑ Homepage (3), the middle area shows the results which match the given search criteria and the right-hand area shows the content of the currently selected object. The search term entered is highlighted in the respective places where it is found in the middle area (see Figure 4-47).

In addition to full text search for a specific search term, filtering of the search results is provided (e.g. by user or edit period). These search criteria can be defined via the "Define Search Criteria" button (see Figure 4-47).

These functions are already familiar from the media selection dialogs introduced with FirstSpirit Version 4.1 (cf. FirstSpirit Release Notes for Version 4.1).

The search dialog in FirstSpirit Version 4.2 is no longer modal; this means that the editor can continue working in JavaClient without closing the search dialog first. This means new options are provided for editorial work. For example, the new search dialog supports Drag & Drop of the search results into JavaClient. In this way, for example, the editor can drag a medium directly from the search dialog into the picture input component of a workspace (see Chapter 4.1.10 page 45).

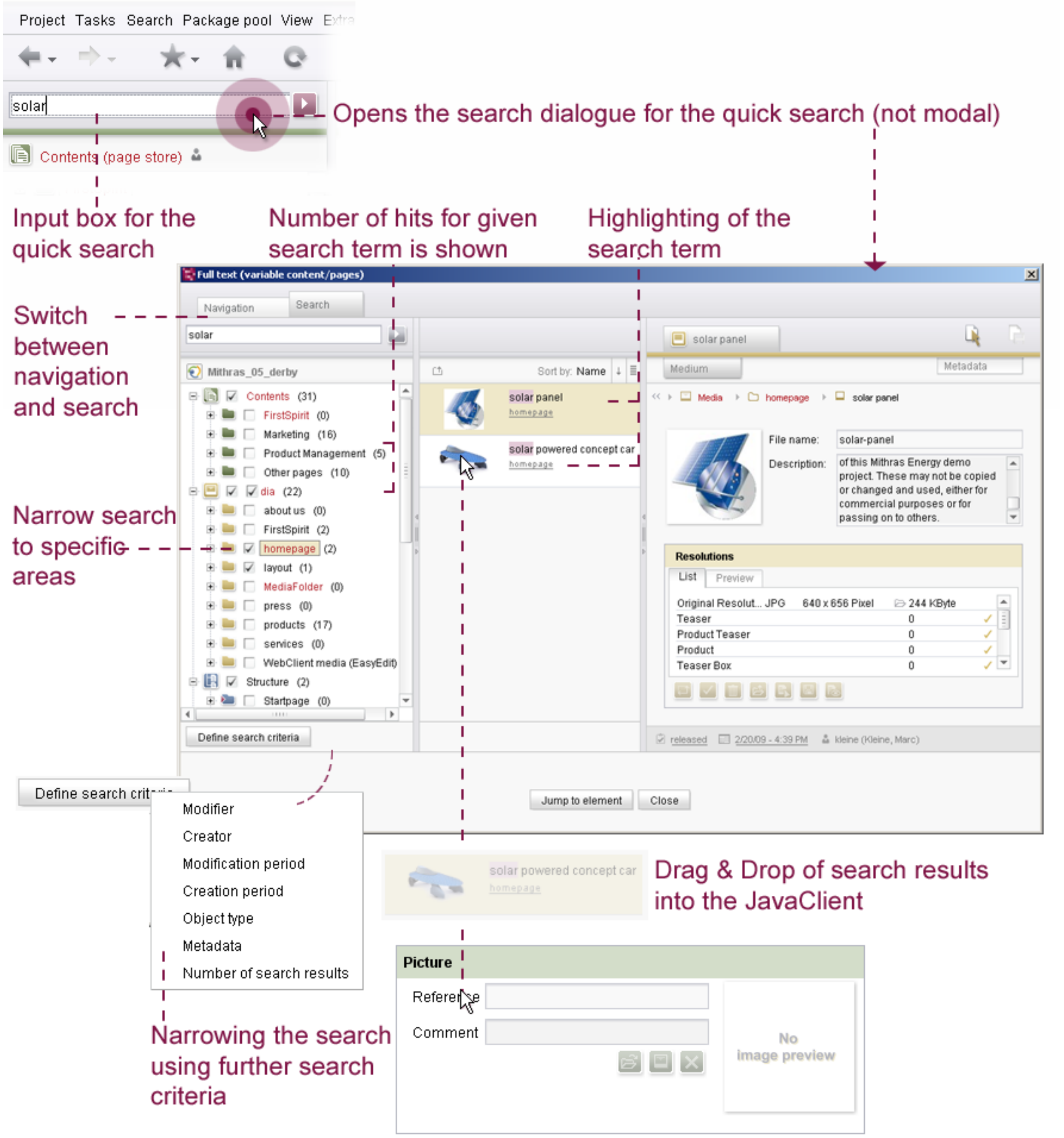
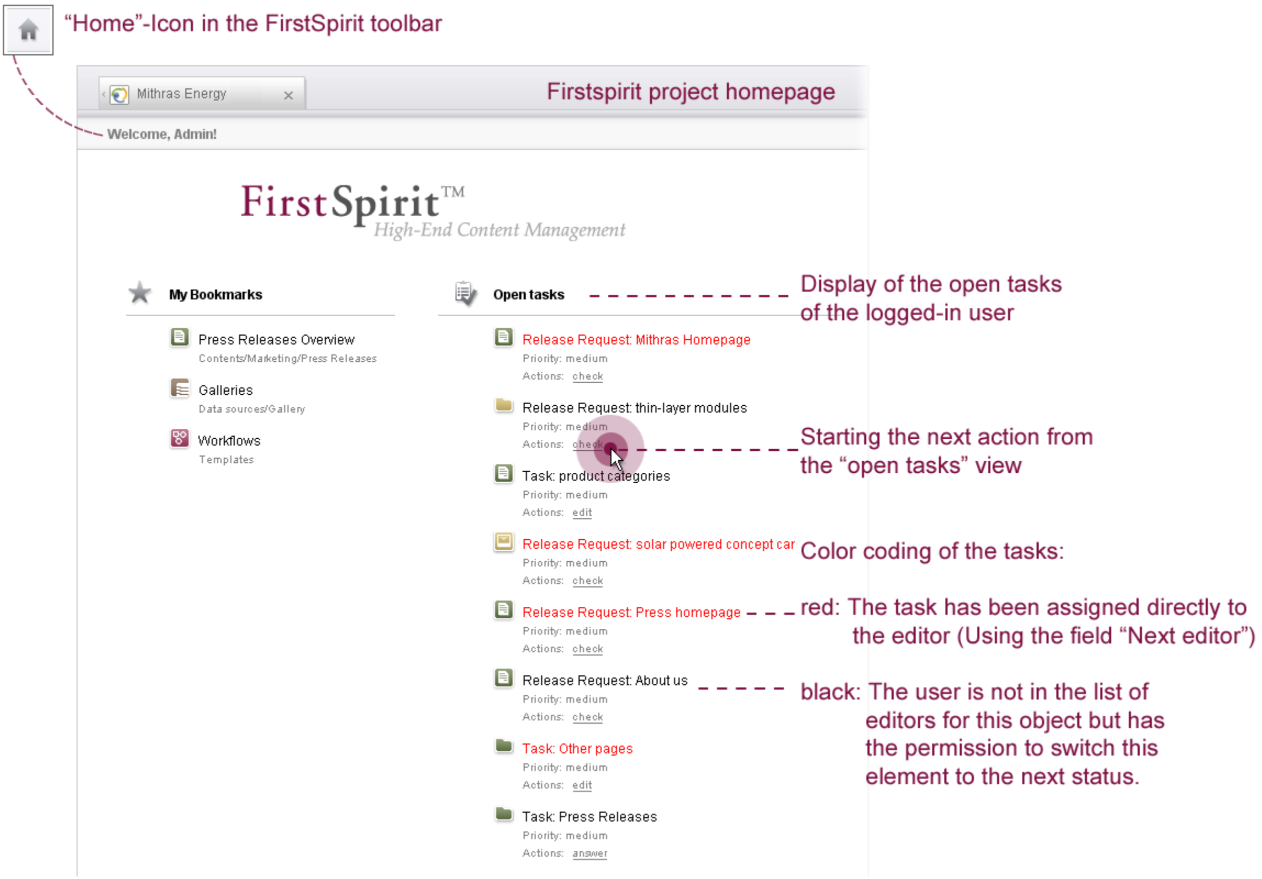

Opens the search dialogue for the quick search (not modal)
Input box for the quick search
Number of hits for given search term is shown
Highlighting of the search term
Switch between navigation and search
Narrow search to specific areas
Define search criteria
Modifier
Creator
Modification period
Creation period
Object type
Metadata
Number of search results
Narrowing the search using further search criteria
Drag & Drop of search results into the JavaClient
Full text (variable content/pages)
Navigation
Search
solar
Jump to element
Close
Picture
Reference
Comment
No image preview

Figure 4-47: New search dialog for quick text search

The "Quick Text Search" function is purely a JavaClient function and is ***not*** *supported in WebEdit (see Chapter 4.3 page 74).*

4.1.16 Open tasks on the FirstSpirit project homepage



“Home”-Icon in the FirstSpirit toolbar

Firstspirit project homepage

Mithras Energy

Welcome, Admin!

FirstSpirit™ High-End Content Management

My Bookmarks

- Press Releases Overview – Contents/Marketing/Press Releases
- Galleries – Data sources/Gallery
- Workflows – Templates

Open tasks

- Release Request: Mithras Homepage – Priority: medium – Actions: check
- Release Request: thin-layer modules – Priority: medium – Actions: check
- Task: product categories – Priority: medium – Actions: edit
- Release Request: solar powered concept car – Priority: medium – Actions: check
- Release Request: Press homepage – Priority: medium – Actions: check
- Release Request: About us – Priority: medium – Actions: check
- Task: Other pages – Priority: medium – Actions: edit
- Task: Press Releases – Priority: medium – Actions: answer

Display of the open tasks of the logged-in user

Starting the next action from the “open tasks” view

Color coding of the tasks:

red: The task has been assigned directly to the editor (Using the field “Next editor”)

black: The user is not in the list of editors for this object but has the permission to switch this element to the next status.

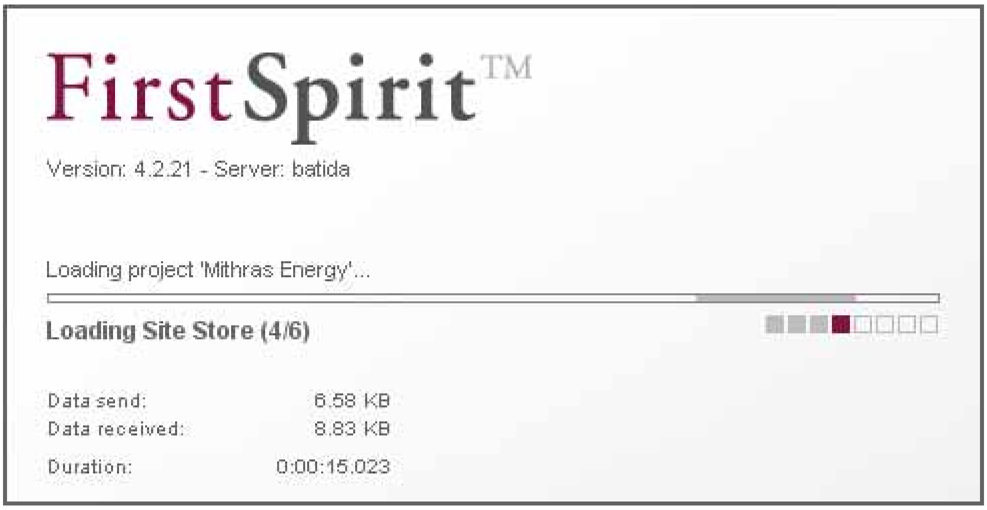
Figure 4-48: Project homepage- New colour coding

The familiar colour concept of the "Open Tasks" task list has now been implemented in the FirstSpirit project homepage too. Since FirstSpirit Version 4.1, on the project homepage, the editor has access to their personal bookmarks and an overview of outstanding – i.e. still to be completed tasks – within the project. The colour concept differentiates between tasks which have been explicitly assigned to the editor – these are displayed with red lettering – and tasks which the editor can open (because of the permissions assigned for the transition), but for which they have not been explicitly entered as the modifier – these tasks are displayed with black lettering.

The "Project Homepage" function is purely a JavaClient function and is not supported in WebEdit (see Chapter 4.3 page 74).

4.1.17 New Start dialog for JavaClient

The Start dialog of the FirstSpirit JavaClient has also been revised within the scope of the changeover to the new look & feel. Among other things, it contains information about a project's loading progress, the quantity of data sent and received and the duration of the start process to date.



FirstSpirit™

Version: 4.2.21 - Server: batida

Loading project 'Mithras Energy'...

Loading Site Store (4/6)

Data send: 6.58 KB
Data received: 8.83 KB
Duration: 0:00:15.023

Figure 4-49: New Start dialog for JavaClient

The "Start Dialog" function is purely a JavaClient function and is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.18 New keyboard shortcuts (CMS_INPUT_DOM / DOMTABLE and version history)

Input component CMS_INPUT_DOM / CMS_INPUT_DOMTABLE: New keyboard shortcuts have been implemented within the CMS_INPUT_DOM input component for fast implementation of actions using the keyboard.

- Ctrl + Backspace: Deletes a word up to the start of the word
- Ctrl + Del: Deletes a word up to the end of the word

The "New Keyboard Shortcuts for CMS_INPUT_DOM" function is also supported in WebEdit (see Chapter 4.3 page 74).

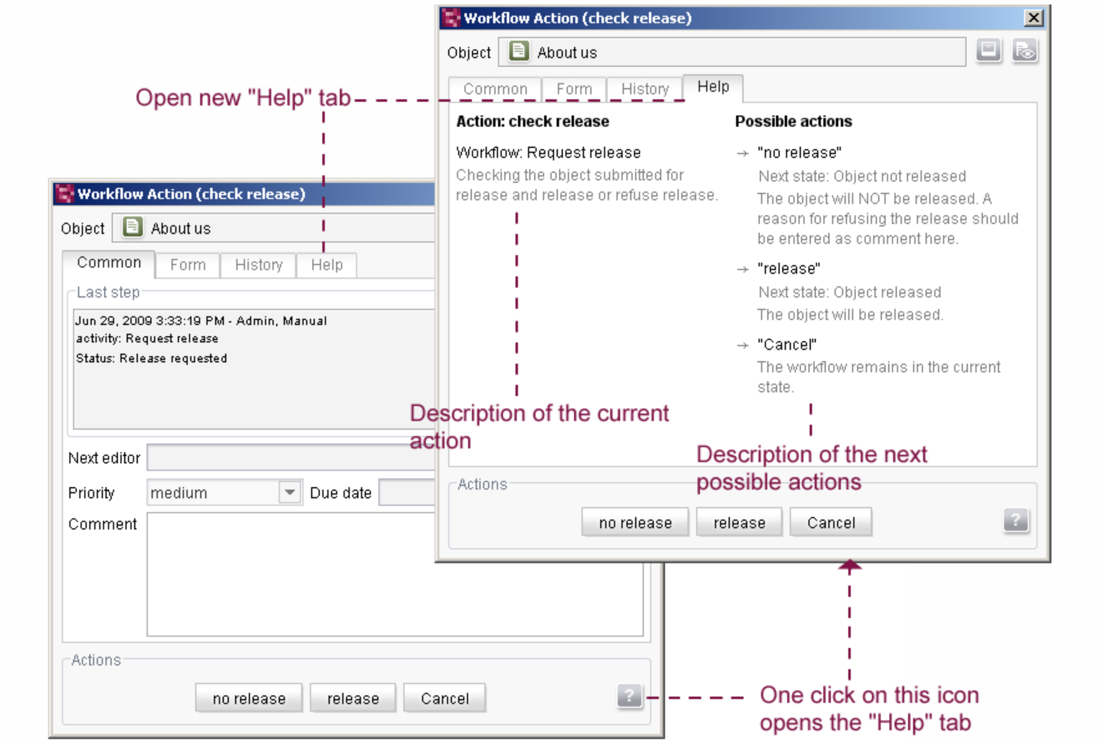
Version history: The version history can be opened on an element using the keyboard shortcut Ctrl + H.

The new keyboard shortcut for opening the version history is purely a JavaClient function and is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.19 User support on connecting workflows

Connecting workflows is supported in FirstSpirit Version 4.2 by a new Help function.

The editor can use the ? button within the action dialog to request further information on the action button in the bottom part of the dialog. The Help tab opens with a description of the current action (in the left-hand part of the dialog) and a description of the possible actions and their consequential states (in the right-hand part of the dialog):


Open new "Help" tab

Workflow Action (check release)

Object: About us

Last step
Jun 29, 2009 3:33:19 PM - Admin, Manual
activity: Request release
Status: Release requested

Priority: medium

Workflow Action (check release)

Object: About us

Action: check release

Workflow: Request release
Checking the object submitted for release and release or refuse release.

Possible actions

→ "no release"
Next state: Object not released
The object will NOT be released. A reason for refusing the release should be entered as comment here.

→ "release"
Next state: Object released
The object will be released.

→ "Cancel"
The workflow remains in the current state.

Description of the current action

Description of the next possible actions

One click on this icon opens the "Help" tab

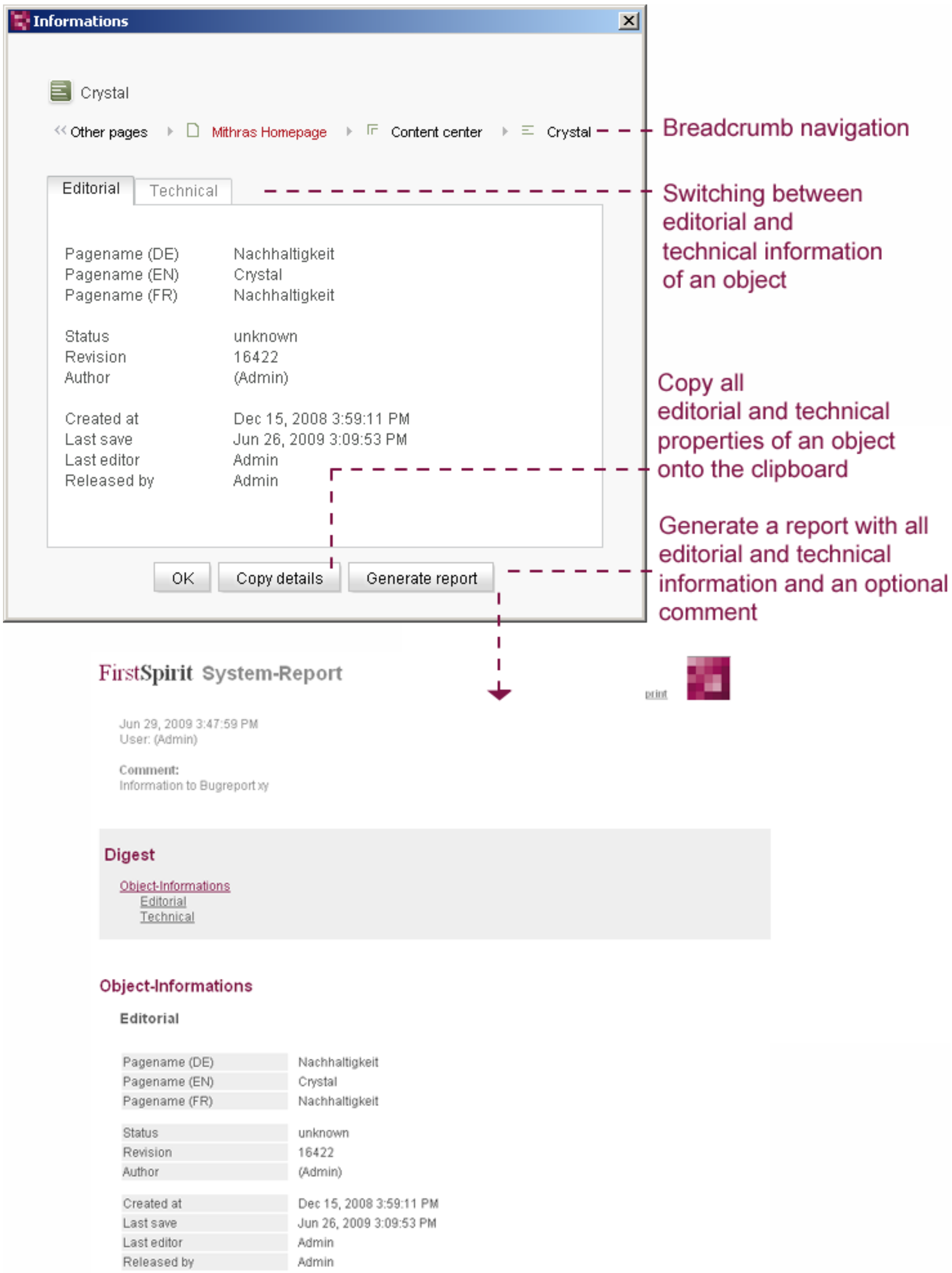
The descriptive texts of the Help dialog can be deposited by the template developer using the language-dependent description within the workflow model (see Chapter 5.1.5 page 83).

The dialogs for starting and switching a multiple selection also support the new Help function.

The "Use Support on Switching Workflows" function is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

4.1.20 Show properties of an object (keyboard shortcut Alt + P)

Using the keyboard shortcut Alt + P information about single project contents could already been displayed in a separate dialog in former FirstSpirit versions. This function has been expanded in FirstSpirit Version 4.2 noticeably. Using the context menu „Extras – Show properties“ or the alternative keyboard shortcut Alt + P the properties of an object selected in the tree view can be displayed.



Pagename (DE)	Nachhaltigkeit
Pagename (EN)	Crystal
Pagename (FR)	Nachhaltigkeit
Status	unknown
Revision	16422
Author	(Admin)
Created at	Dec 15, 2008 3:59:11 PM
Last save	Jun 26, 2009 3:09:53 PM
Last editor	Admin
Released by	Admin

Breadcrumb navigation

Switching between editorial and technical information of an object

Copy all editorial and technical properties of an object onto the clipboard

Generate a report with all editorial and technical information and an optional comment

FirstSpirit System-Report

Jun 29, 2009 3:47:59 PM
User: (Admin)

Comment:
Information to Bugreport xy

Digest

Object-Informations
Editorial
Technical

Object-Informations

Editorial

Pagename (DE)	Nachhaltigkeit
Pagename (EN)	Crystal
Pagename (FR)	Nachhaltigkeit
Status	unknown
Revision	16422
Author	(Admin)
Created at	Dec 15, 2008 3:59:11 PM
Last save	Jun 26, 2009 3:09:53 PM
Last editor	Admin
Released by	Admin

Figure 4-50: Information dialog with generated system report

The dialog is separated into editorial and technical information. They are shown when switching to the corresponding tab:

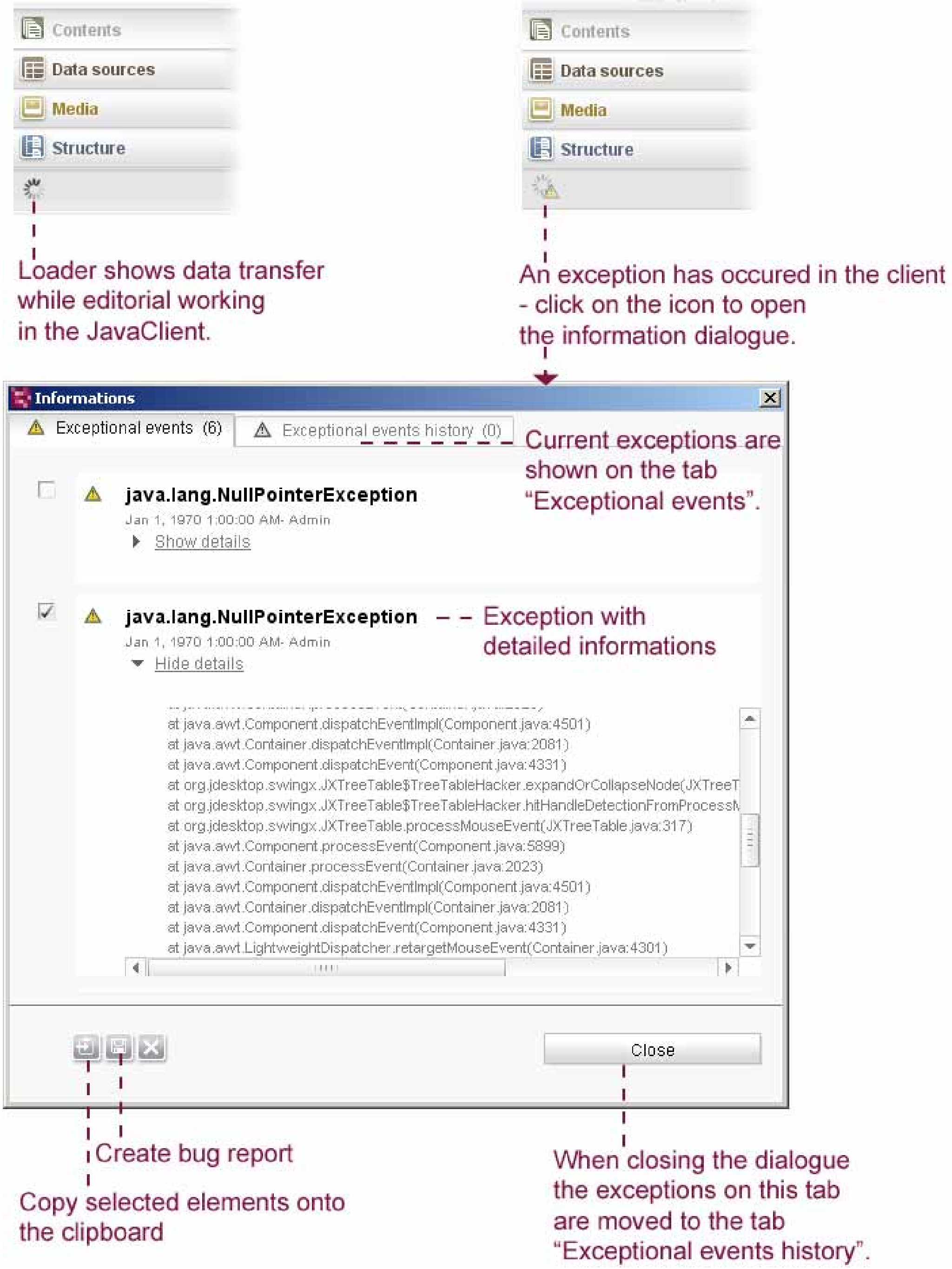
- **Editorial:** This tab shows the object's properties which are relevant for the editor. This includes for example the language dependent display name, the status and revision of an object, the author, the last editor and the time of creation and last saving the object.
- **Technical:** This tab shows the technically relevant properties of an object. This includes for example the unique reference name and the unique ID of an object and the name and the ID of the related template. Moreover, the complete path (label path, UID path, ID path, template path) of the object to the respective Store root is displayed (this path information can also be displayed using the keyboard shortcut Ctrl + Shift + Q).
- **Copy details:** Using this button all displayed information (editorial and technical) can be copied to the clipboard.
- **Generate report:** Using this button a system report can be generated containing the displayed information (editorial and technical) (see Figure 4-50). A comment, e.g. a bug description, can be added to the report, too.

The function "Show properties of an object" is purely a JavaClient function and is not supported in WebEdit (see Chapter 4.3 page 74).

4.1.21 Central collection of bugs and system report

With FirstSpirit Version 4.2 a new infrastructure is provided for collecting bugs and exceptions.

For this purpose, a loader icon is displayed in the bottom left area of the JavaClient which indicates continuously the data transfer while the doing editorial work. If an exception occurs a small exclamation mark will be displayed in the icon. Click on this icon to get further information about the occurred bug. An information dialog will open with an overview of the current exception (tab "Exceptional event"). Click on the link "Show details" to open the complete stacktrace of the exception.



Loader shows data transfer while editorial working in the JavaClient.

An exception has occured in the client - click on the icon to open the information dialogue.

Current exceptions are shown on the tab “Exceptional events”.

□ java.lang.NullPointerException
Jan 1, 1970 1:00:00 AM- Admin

☑ java.lang.NullPointerException
Jan 1, 1970 1:00:00 AM- Admin

Exception with detailed informations

```
at java.awt.Component.dispatchEventImpl(Component.java:4501)
at java.awt.Container.dispatchEventImpl(Container.java:2081)
at java.awt.Component.dispatchEvent(Component.java:4331)
at org.jdesktop.swingx.JXTreeTable$TreeTableHacker.expandOrCollapseNode(JXTreeT
at org.jdesktop.swingx.JXTreeTable$TreeTableHacker.hitHandleDetectionFromProcessM
at org.jdesktop.swingx.JXTreeTable.processMouseEvent(JXTreeTable.java:317)
at java.awt.Component.processEvent(Component.java:5899)
at java.awt.Container.processEvent(Container.java:2023)
at java.awt.Component.dispatchEventImpl(Component.java:4501)
at java.awt.Container.dispatchEventImpl(Container.java:2081)
at java.awt.Component.dispatchEvent(Component.java:4331)
at java.awt.LightweightDispatcher.retargetMouseEvent(Container.java:4301)
```

Create bug report

Copy selected elements onto the clipboard

When closing the dialogue the exceptions on this tab are moved to the tab “Exceptional events history”.

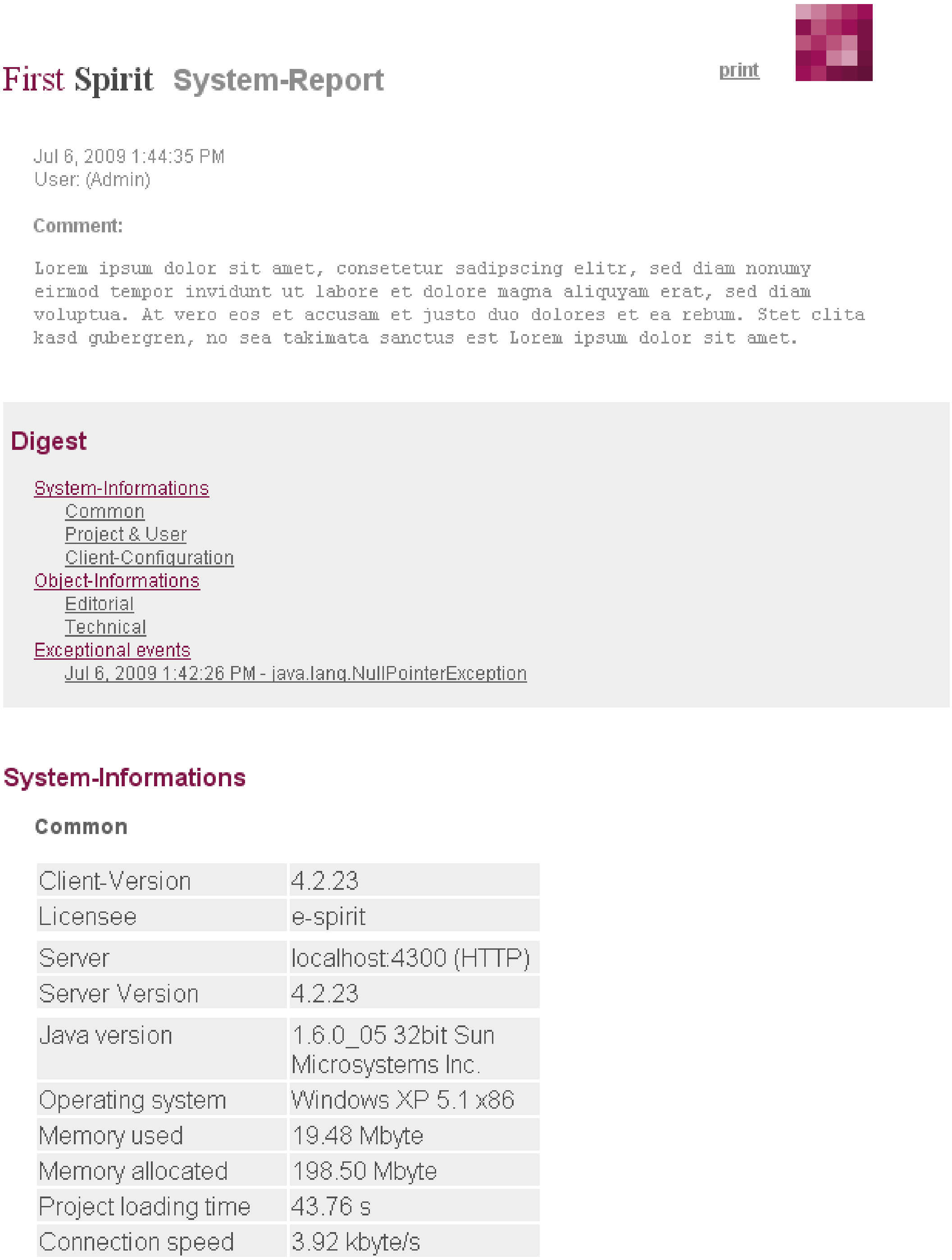
Figure 4-51: Central bug collection in the FirstSpirit JavaClient

One or more error messages can be selected in the information dialog and copied to the clipboard and/or subsumed in a bug report (see Chapter 4.1.20 page 63).

The bug report contains, in addition to the information about the exception, further information which can be relevant for reproducing the bug:

- project settings,
- configuration of the operating system and the server,

- JavaClient configuration (e.g. configuration of the integrated preview and the used browser engine),
- Object (technical and editorial information)(see Chapter 4.1.20 page 63)



First Spirit System-Report

print

Jul 6, 2009 1:44:35 PM
User: (Admin)

Comment:

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy
eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam
voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita
kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Digest

- System-Informations
  - Common
  - Project & User
  - Client-Configuration
- Object-Informations
  - Editorial
  - Technical
- Exceptional events
  - Jul 6, 2009 1:42:26 PM - java.lang.NullPointerException

System-Informations

Common

Client-Version	4.2.23
Licensee	e-spirit
Server	localhost:4300 (HTTP)
Server Version	4.2.23
Java version	1.6.0_05 32bit Sun Microsystems Inc.
Operating system	Windows XP 5.1 x86
Memory used	19.48 Mbyte
Memory allocated	198.50 Mbyte
Project loading time	43.76 s
Connection speed	3.92 kbyte/s

Figure 4-52: Generated system report (extract)

The exception remains available within the actual user session even if the error dialog is closed and will be shown on the tab “Exceptional events history” when the information dialog will be opened the next time.

The function “Central collection of bugs and system report” is purely a JavaClient function and is not supported in WebEdit (see Chapter 4.3 page 74).

4.2 New functions in WebClient

4.2.1 Re-design of the WebClient

The WebClient has been comprehensibly redesigned for WebEdit 4.2. To this end (among other things), the "xp" theme, which can be configured for a project by the project administrator, has been revised:

- Revised toolbar with new buttons (see Figure 4-53)
- Adjustment of the layout within selection dialogs

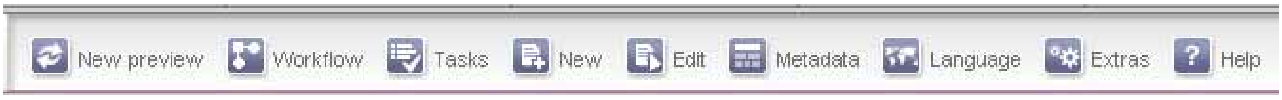
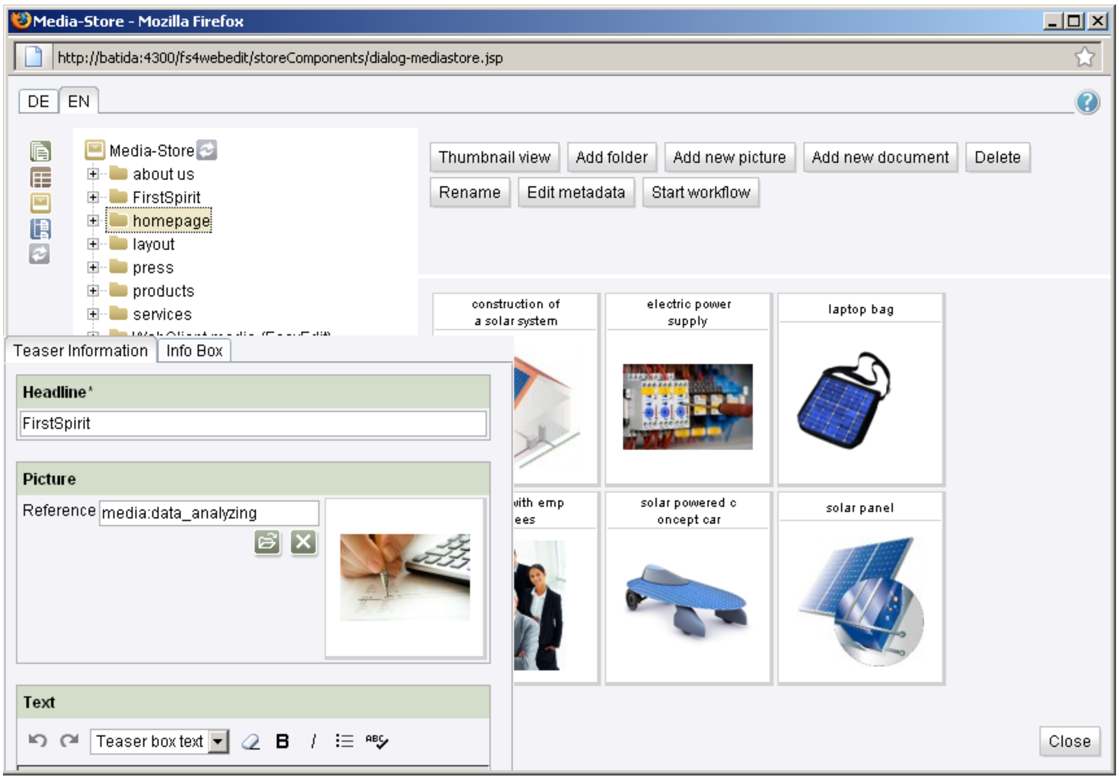


Figure 4-53: WebEdit toolbar

In addition, several enhancements have been implemented within the input components and selection dialogs, for example, the:

- Preview within picture input components (see Figure 4-54)
- Display of elements in the Media Store (see Figure 4-54)
- Fading effects on loading new content (e.g. on generating a preview)
- Additional controls can be displayed in the preview using layer and iFrames (see Figure 4-56)



http://batida:4300/fs4webedit/storeComponents/dialog-mediastore.jsp

Headline* FirstSpirit

Reference media:data_analyzing

Figure 4-54: WebClient re-design

The project selection dialog, login and errors page for WebEdit 4.2 have also been changed within the scope of the redesign.

4.2.2 Easy-Edit

The "Easy-Edit" function was introduced to enable direct editing of editorial content within the preview page without using separate windows (pop-up windows). It can therefore replace the Quick Edit function at section level. The editor works directly in the project's preview in the browser where they can edit defined areas ("Hotspots") of the page. New sections can also be inserted directly in the required position by pressing a button.

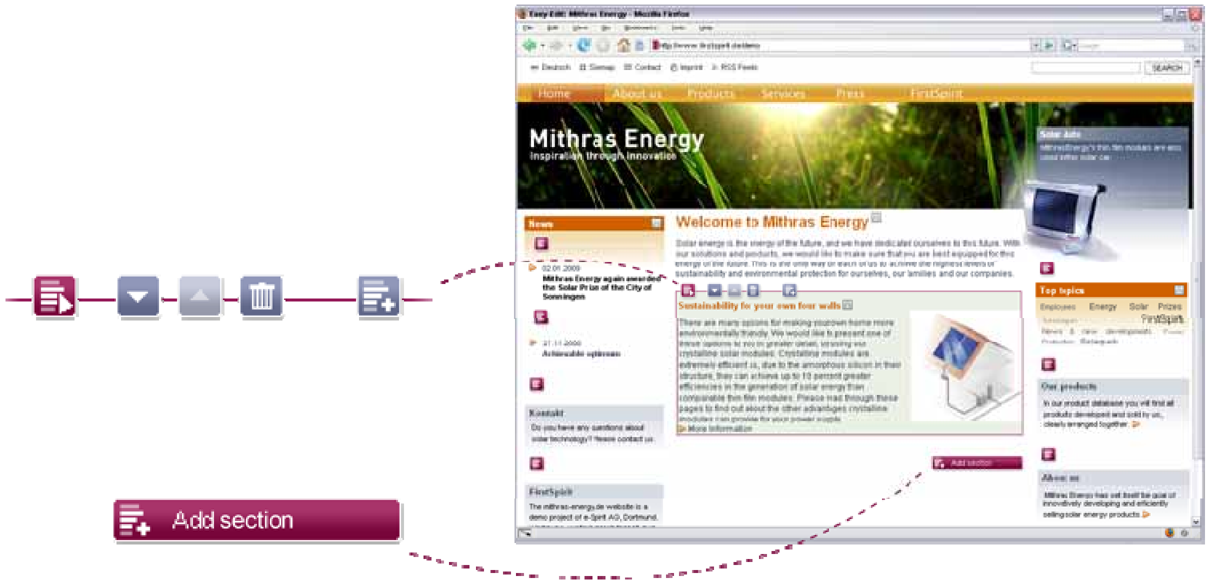


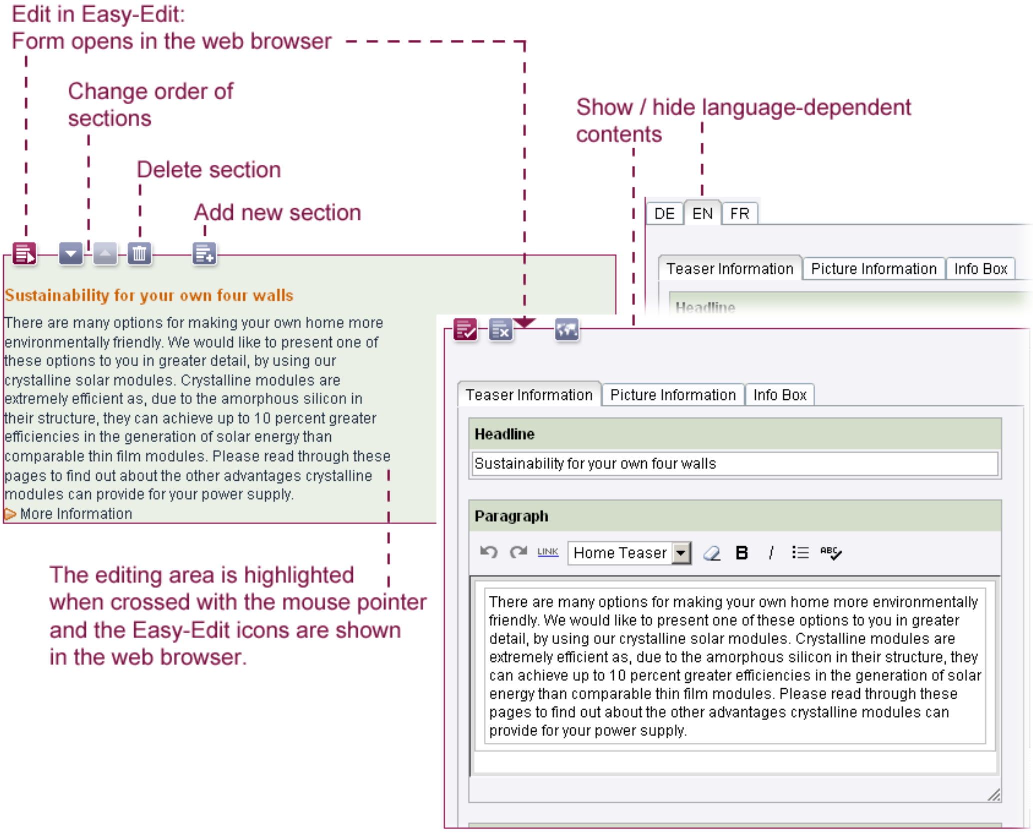
Figure 4-55: "Easy-Edit" in WebEdit 4.2

The "Easy-Edit areas", in which the editor can enter or change content, are highlighted in colour and by a frame on the preview page when the mouse is passed over them. In addition, icons are displayed and available for the editing (see Figure 4-56).

The Easy-Edit function is available at section and at page level, as well as for the maintenance of structured data from the Content Store of FirstSpirit. The Easy-Edit editing options also change depending on the editable area:

- Page content: Can be edited using Easy-Edit.
- Section content: Can be edited using Easy-Edit. In addition, it is possible to move or to delete the whole section and to create new sections (see Figure 4-56).
- Database content (from the Content Store): Can be edited using Easy-Edit. In addition, it is possible to remove the whole dataset and to create new datasets (see Figure 4-57).

If the editor clicks the button for editing, the editing dialog with the input components defined for the respective area are displayed directly above the preview page. Another icon ("Language Display") can be used to show and hide language-dependent editorial content as registers in the editing window and can therefore also be quickly changed if necessary.



Edit in Easy-Edit:
Form opens in the web browser

Change order of sections

Delete section

Add new section

Show / hide language-dependent contents

The editing area is highlighted when crossed with the mouse pointer and the Easy-Edit icons are shown in the web browser.

Sustainability for your own four walls

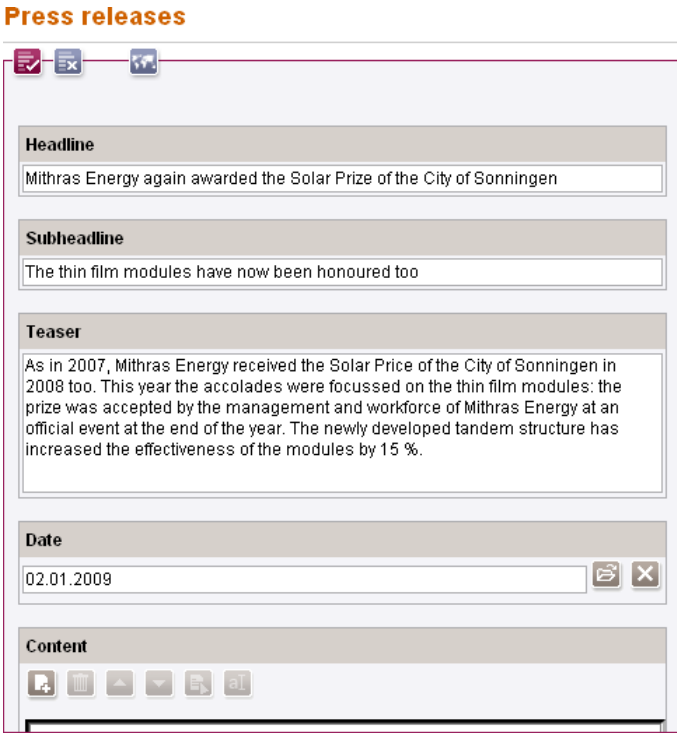
There are many options for making your own home more environmentally friendly. We would like to present one of these options to you in greater detail, by using our crystalline solar modules. Crystalline modules are extremely efficient as, due to the amorphous silicon in their structure, they can achieve up to 10 percent greater efficiencies in the generation of solar energy than comparable thin film modules. Please read through these pages to find out about the other advantages crystalline modules can provide for your power supply.

Headline: Sustainability for your own four walls

Paragraph: There are many options for making your own home more environmentally friendly. We would like to present one of these options to you in greater detail, by using our crystalline solar modules. Crystalline modules are extremely efficient as, due to the amorphous silicon in their structure, they can achieve up to 10 percent greater efficiencies in the generation of solar energy than comparable thin film modules. Please read through these pages to find out about the other advantages crystalline modules can provide for your power supply.

Figure 4-56: Editing a section in Easy-Edit

The input components within the form are displayed in the colour concept of the respective Store, in Figure 4-56 in the colouring of the Page Store, in Figure 4-57 in the colouring of the Content Store.



Press releases

Headline
Mithras Energy again awarded the Solar Prize of the City of Sonningen

Subheadline
The thin film modules have now been honoured too

Teaser
As in 2007, Mithras Energy received the Solar Price of the City of Sonningen in 2008 too. This year the accolades were focussed on the thin film modules: the prize was accepted by the management and workforce of Mithras Energy at an official event at the end of the year. The newly developed tandem structure has increased the effectiveness of the modules by 15 %.

Date
02.01.2009

Content

Figure 4-57: Editing content from the Content Store

In order to use the "Easy-Edit" function, the project's templates must first be adjusted by the template developer (see Chapter 5.1.6 page 85).

4.2.3 Generic link editors

In FirstSpirit Version 4.2, the configuration options for links are considerably enhanced by the introduction of generic link editors (see Chapter 4.1.7 page 41).

However, there are restrictions to use of the corresponding new input components in WebEdit (see Chapter 5.1.9 page 88).

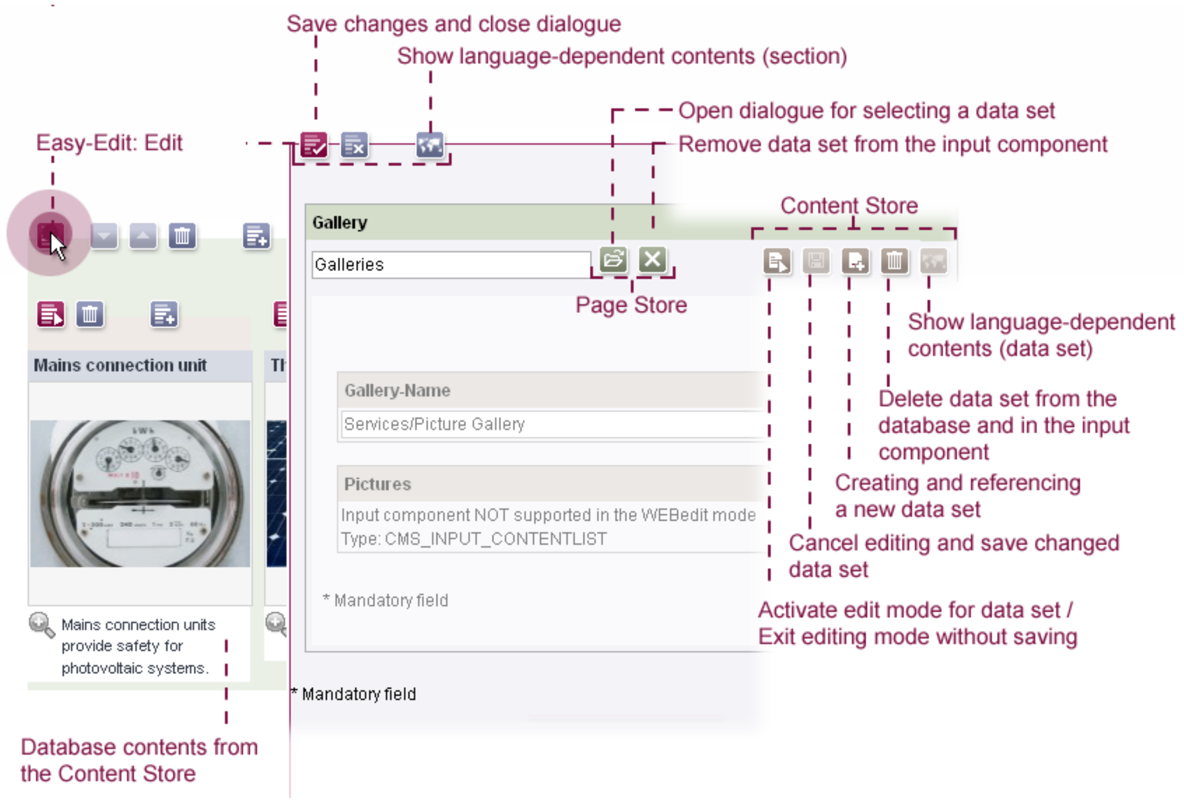
- Restrictions to use of CMS_INPUT_OBJECTCHOOSER and FS_DATASET (see Chapter 4.2.4 page 71)

4.2.4 CMS_INPUT_OBJECTCHOOSER enhancement

The enhancements of the CMS_INPUT_OBJECTCHOOSER input component are also supported in FirstSpirit WebClient. But the new functions are slightly restricted compared to their use FirstSpirit JavaClient (cf. Chapter 4.1.12 page 54):

- For example, the option of switching to the Content Store with the click of a button in order to edit the selected dataset ("Jump to...") is not available in WebEdit.

- The other uses of a referenced dataset can also not be displayed in WebClient.
- In addition, the layout of the components in WebEdit and the logic of the buttons for saving and editing the input is somewhat different (see Figure 4-58):


Save changes and close dialogue
Show language-dependent contents (section)
Open dialogue for selecting a data set
Remove data set from the input component
Easy-Edit: Edit
Content Store
Gallery
Page Store
Show language-dependent contents (data set)
Mains connection unit
Gallery-Name
Delete data set from the database and in the input component
Creating and referencing a new data set
Pictures
Cancel editing and save changed data set
Activate edit mode for data set / Exit editing mode without saving
Database contents from the Content Store

Galleries

Services/Picture Gallery

Input component NOT supported in the WEBedit mode
Type: CMS_INPUT_CONTENTLIST

* Mandatory field

Mains connection units provide safety for photovoltaic systems.

* Mandatory field

Figure 4-58: Enhancements to the object chooser input component

In order to accept the changed content in the referenced dataset, the dataset change must be explicitly saved in the WebEdit input component (see Figure 4-58 "Save changes to dataset"). Simply saving the change using the form ("Save changes and close dialog") as in FirstSpirit JavaClient is not sufficient in WebEdit.

All other functions can be used analogous to FirstSpirit JavaClient (see Chapter 4.1.12 page 54).

<u>**Note:**</u> Nested use of the CMS_INPUT_OBJECTCHOOSER and FS_DATASET input components is not supported in WebEdit. This means "Object Chooser-in-Object Chooser", "Dataset-in-Object Chooser", "Object chooser-in-Dataset" and "Dataset-in-Dataset" cannot be used.

4.2.5 Support for inline tables

Since FirstSpirit Version 4.1, so-called "inline tables" can be integrated in the text flow by enhancement of the DOM Editor input component (CMS_INPUT_DOM) in JavaClient (see FirstSpirit Release Notes for Version 4.1). The function is now available in WebEdit 4.2 – albeit with limited scope.

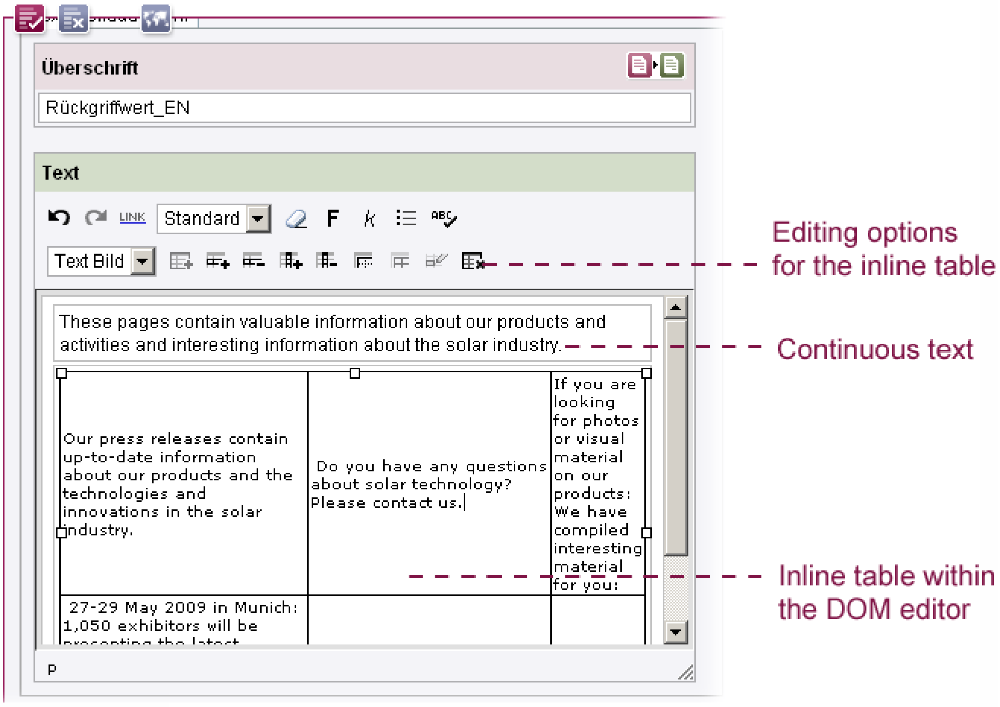
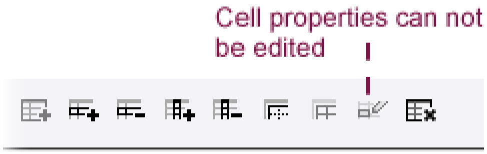

Überschrift
Rückgriffwert_EN
Text
LINK
Standard
F
k
Text Bild
These pages contain valuable information about our products and activities and interesting information about the solar industry.
Our press releases contain up-to-date information about our products and the technologies and innovations in the solar industry.
Do you have any questions about solar technology? Please contact us.
If you are looking for photos or visual material on our products: We have compiled interesting material for you:
27-29 May 2009 in Munich: 1,050 exhibitors will be presenting the latest
P
Editing options for the inline table
Continuous text
Inline table within the DOM editor

Figure 4-59: Inline table within the DOM Editor in WebEdit 4.2

The WebEdit editor can add tables to the DOM Editor using an appropriately configured input component. Buttons for this are available within the editing area (see Figure 4-59). The functions available for editing the added table depend on the predefined settings of the template developer. The following are possible:

- Add / Remove columns
- Add / Remove rows
- Add / Remove tables
- Merge cells / split cells

The table layout is specified by the template developer using table format templates and so-called style templates. Input components for maintenance of layout attributes can be created using a style template (e.g. for background colour, text alignment, font, etc.), which can be assigned specific default values by the template developer ("retrieval values"). The editor cannot change these properties in WebEdit (unlike JavaClient). This means, the button for editing the cell properties is always disabled in WebEdit.



Cell properties can not be edited

Figure 4-60: Restrictions

For further information on the "Inline Table" function, see FirstSpirit Manual for Editors (WebClient) and FirstSpirit Online Documentation.

4.3 Restrictions

<u>The following functions are supported in JavaClient only:</u>

- Multi-tabbing (see Chapter 4.1.1 page 17)
- Breadcrumb navigation (see Chapter 4.1.2 page 23)
- Individual display of the Stores (see Chapter 4.1.3 page 24)
- Integrated preview (see Chapter 4.1.4 page 25)
- Content-highlighting (see Chapter 4.1.5 page 30)
- Media gallery (see Chapter 4.1.6 page 35)
- Optimised display of datasets (see Chapter 4.1.8 page 42)
- Full text search in the Content Store (see Chapter 4.1.9 page 44)
- Drag & Drop FirstSpirit objects (see Chapter 4.1.10 page 45)
- Restore settings on restart (see Chapter 4.1.11 page 53)
- Extension of the reference graph (see Chapter 4.1.13 page 56)
- Status display in the workspace (see Chapter 4.1.14 page 57)
- New quick text search (non-modal) (see Chapter 4.1.15 page 58)
- Open tasks on the project homepage (see Chapter 4.1.16 page 60)
- New Start dialog (see Chapter 4.1.17 page 61)
- Keyboard shortcut for opening version history (see Chapter 4.1.18 page 61)
- User support for workflows (see Chapter 4.1.19 page 62)
- Show properties of an object (see Chapter 4.1.20 page 63)
- Central bug collection and system report (see Chapter 4.1.21 page 64)
- Import of MS Word documents (see Chapter 5.2.1 page 98)
- Language dependent display of content areas (see Chapter 5.1.12 page 96)

<u>The following functions are supported in WebClient and in JavaClient:</u>

- Generic link editors (see Chapter 4.1.7 page 41)
- CMS_INPUT_OBJECTCHOOSER (see Chapter 4.1.12 page 54)
- Access to several remote media projects (see Chapter 5.2.2 page 99)
- Inline tables (see Chapter 4.2.5 page 72)
- New keyboard shortcuts in the DOM Editor (see Chapter 4.1.18 page 61)

The following functions are supported in WebClient only:

▪ Redesign of WebClient	(see Chapter 4.2.1 page 67)
▪ Easy-Edit	(see Chapter 4.2.2 page 68)

5 New Functions for Template Developers

5.1 New functions in JavaClient

5.1.1 Integrated form preview in the Template Store

A template developer can use the integrated preview to check changes in the output channel of the templates (e.g. HTML or XSL-FO) directly in the preview window, as each time the template is saved, the (configured) preview page is automatically updated.

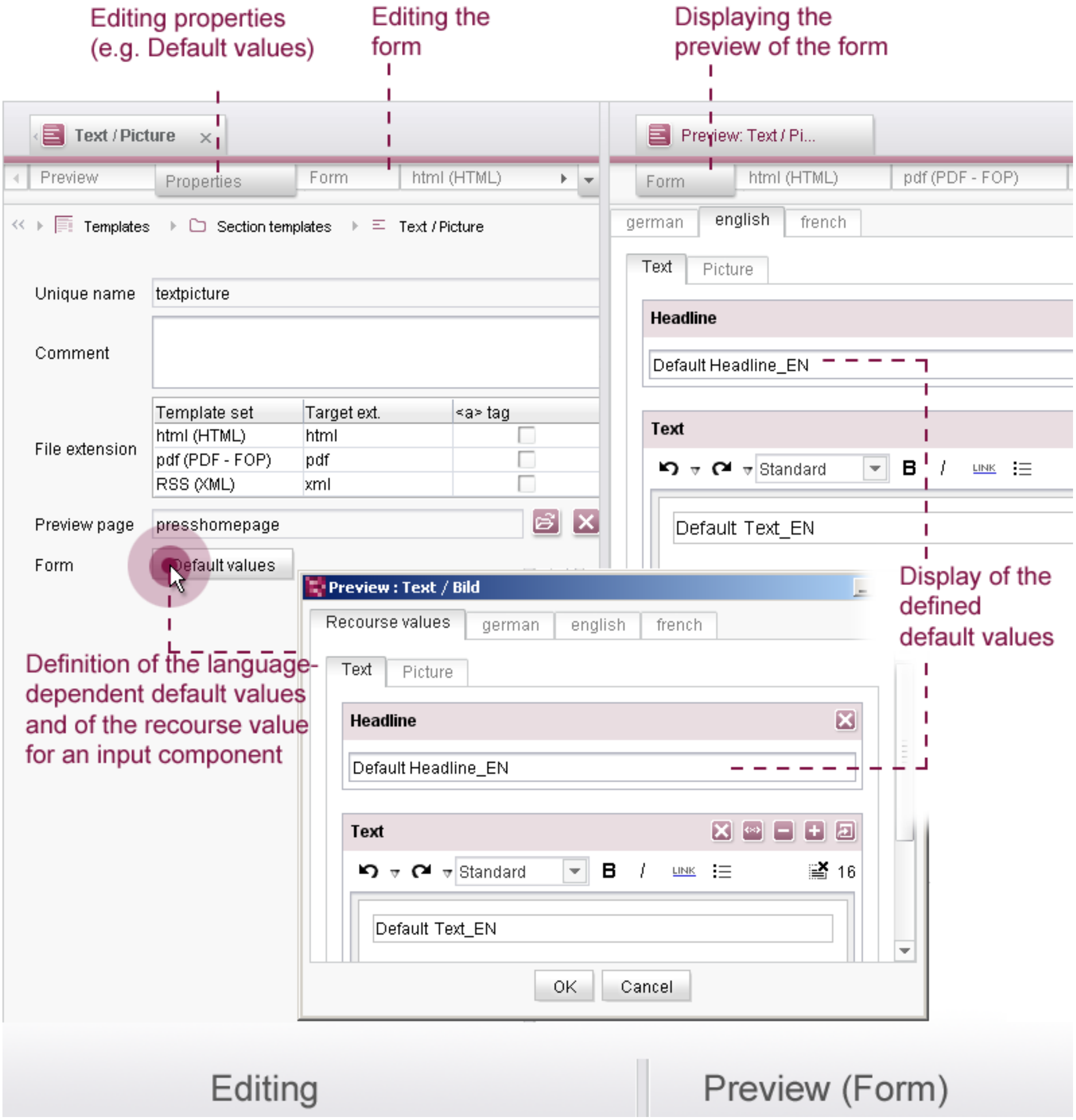
In addition, an integrated form preview is available within the Template Store. If the form area of a template is selected, a live view of the edited form appears in the preview area with the defined retrieval values of the input components (see Figure 5-1).

The integrated preview can be optionally displayed on the right next to the workspace or, if smaller monitors are used, in an external window.

All input fields can be directly edited within the integrated form preview. To do this, the template must not be locked to prevent editing.

Note: The editing option is merely a tool for the template developer. In addition, for example, it is also possible to directly check whether a defined remote configuration provides the required results for an input component. However, the content entered is not saved in the form view. Default values cannot be defined in the form preview.

The **default values for the input components** can be defined using the "Default Values" button in the "Properties" tab of a template. The language-dependent default values are displayed in the preview window immediately after saving the properties.



Editing properties (e.g. Default values)

Editing the form

Displaying the preview of the form

Display of the defined default values

Definition of the language-dependent default values and of the recourse value for an input component

Unique name: textpicture

Comment:

File extension:

Template set	Target ext.	<a> tag
html (HTML)	html	□
pdf (PDF - FOP)	pdf	□
RSS (XML)	xml	□

Preview page: presshomepage

Form: Default values

Headline: Default Headline_EN

Text: Default Text_EN

Figure 5-1: Preview of the form area in the Template Store

For further information on the Form Preview, see "FirstSpirit Manual for Developers" (Basics).

5.1.2 Generic link editors

Editorial maintenance of links in FirstSpirit is carried out using screen forms which can be adjusted, project-specifically, by the template developer. The forms are based on link configurations which specify in detail, which input fields are available to the editor for the maintenance of links. To date, configuration of these static editors has merely involved showing or hiding the pre-defined fields, and an option for changing the default lettering of these fields.

From FirstSpirit Version 4.2, the configuration options are considerably enhanced by the introduction of generic link editors. The configuration of link templates can then, analogous to page and section templates, be created by adding input components into the form area. In this way, on creating a link configuration, template developers can fall back on all FirstSpirit input components and therefore adapt the input options of the editors far more project-specifically than before.

All input options for the maintenance of links can be reproduced **using the normal form syntax of FirstSpirit**.

As part of the introduction of generic link editors, the link templates can now also be structured in folders.

The conventional input options for links (of the static link editors) can of course also be generated using the new, generic editors. Several new input components were introduced with FirstSpirit Version 4.2, to enable all the functions of the static link editors used to date to be reproduced on the new generic editors (see Chapter 5.1.9 page 88). For example, selection via the "mediaref" field of the static link editors could not be reproduced on the existing input components of FirstSpirit Version 4.1. The input components CMS_INPUT_PICTURE and CMS_INPUT_FILE each only support selection of one reference type, i.e. either pictures or files, but not both. Therefore, in FirstSpirit Version 4.2, the input component FS_REFERENCE was introduced, which supports any reference types (see Figure 5-8).

Analogous to this, enhancements to the input component CMS_INPUT_OBJECTCHOOSER for reproduction of a link on database content, for selection of datasets from *one* specific database table (see Chapter 4.1.12 page 54) and the new input component FS_DATASET, for selection of datasets from *any* database tables, were introduced (see Figure 5-9).

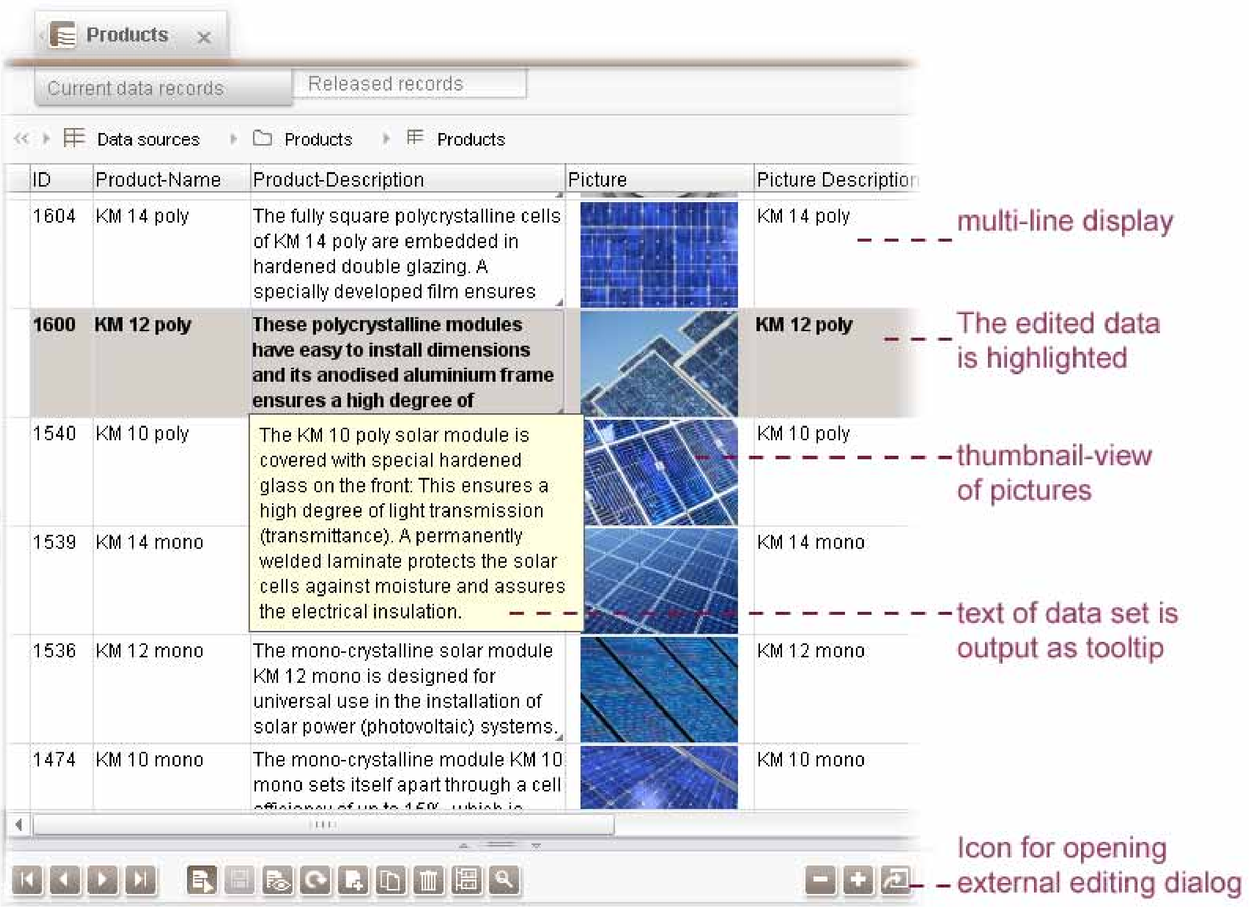
Note: Use of the new components is subject to certain limitations, as the components will not be officially released until FirstSpirit Version 5.0 (see Chapter 5.1.9 page 88).
In FirstSpirit Version 4.2, the new components are mainly used for migration of the existing static editors (see Chapter 8.1.2 page 114). Further use of the components is possible, if the project developers dispense with use of the API and they are willing to potentially subsequently adjust the parameter assignment of the input components (see Chapter 5.1.9 page 88).

Note: The new functions and enhancements in FirstSpirit Version 4.2 are intended to prepare existing projects for the new major release of FirstSpirit Version 5.0. The new concepts introduced can still be used in FirstSpirit Version 4.2 in parallel with the functions used to date (in Version 4.1). With the release of FirstSpirit Version 5.0, individual functions will then no longer be supported, but will instead be replaced by new, enhanced concepts. An example is the introduction of generic link editors in FirstSpirit Version 4.2. These provide greater functional scope and more flexible design options than the static editors used to date. **The static link editors will no longer be supported from FirstSpirit Version 5.0.** The introduction of "generic link editors" in FirstSpirit Version 4.2 therefore also helps to migrate existing projects to FirstSpirit Version 5.0 (see Chapter 8.1.2 page 114).

The "Generic Link Editors" function is also supported in WebEdit (see Chapter 4.3 page 74).

5.1.3 Configurable, multi-line display of datasets

With FirstSpirit Version 4.2, the display of content within the Content Store should be optimised for editorial work. To this end, options for configuring the views are made available within the Template Store:



ID	Product-Name	Product-Description	Picture	Picture Descriptior
1604	KM 14 poly	The fully square polycrystalline cells of KM 14 poly are embedded in hardened double glazing. A specially developed film ensures		KM 14 poly
1600	**KM 12 poly**	**These polycrystalline modules have easy to install dimensions and its anodised aluminium frame ensures a high degree of**		**KM 12 poly**
1540	KM 10 poly			KM 10 poly
1539	KM 14 mono			KM 14 mono
1536	KM 12 mono	The mono-crystalline solar module KM 12 mono is designed for universal use in the installation of solar power (photovoltaic) systems.		KM 12 mono
1474	KM 10 mono	The mono-crystalline module KM 10 mono sets itself apart through a cell		KM 10 mono

The KM 10 poly solar module is covered with special hardened glass on the front: This ensures a high degree of light transmission (transmittance). A permanently welded laminate protects the solar cells against moisture and assures the electrical insulation.

multi-line display

The edited data is highlighted

thumbnail-view of pictures

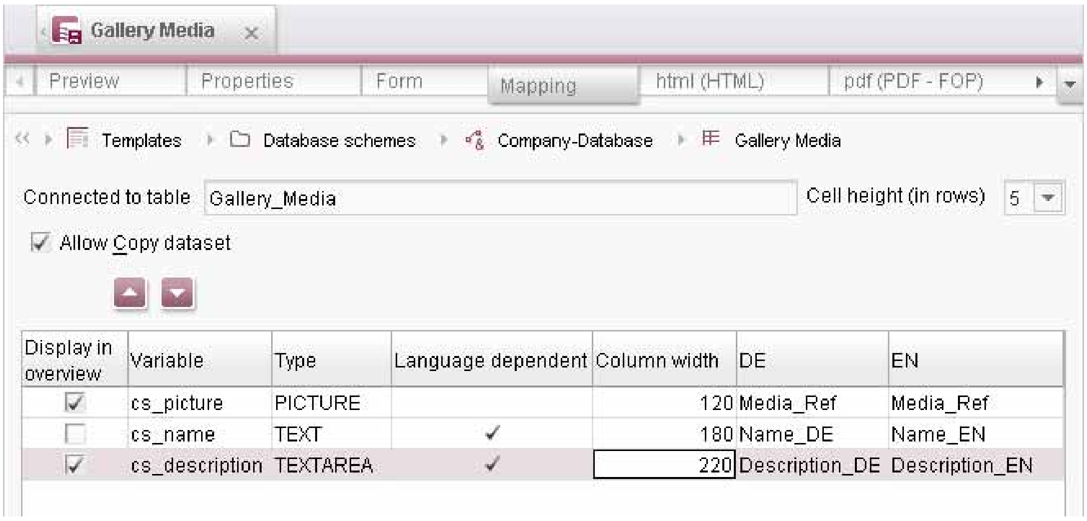
text of data set is output as tooltip

Icon for opening external editing dialog

Figure 5-2: Multi-line display of datasets

- **Configurable, multi-line display of datasets:** Apart from the familiar single line view of datasets, depending on the configuration of the table templates, multiple column views can now also be configured. These multi-line displays use the available area for multi-line text outputs and the display of pictures as thumbnails (see Figure 5-2). The familiar single line display can continue to be used.

- **Text output as tooltips:** If the text to be displayed within an input component is larger than the area for the component available in the overview, the text is inserted as a tooltip when the mouse cursor is passed over it. In the case of very long texts, the text length is shortened within the tooltip. In this way, the editor can find out about the content of a dataset directly within the overview, without opening it first in the detailed view (see Figure 5-2).
- **Display of pictures:** Pictures added using the CMS_INPUT_PICTURE input component are displayed as thumbnails within the multi-line display. The size of the thumbnails results from the line height of the displayed dataset (see Figure 5-2).
- **CMS_INPUT_CONTENTAREALIST:** The content of the CMS_INPUT_CONTENTAREALIST input component is also displayed in the overview.
- **Other configuration options:** Further configuration settings for display of the overview in the Content Store can also be made using the table templates (see Figure 5-3):
  Depending on a setting in the table templates, the *copying of datasets* can now be allowed or prevented. The *layout of the columns* can be changed using a simple *sort function* within the table template. Irrelevant columns can be hidden from the overview using a checkbox within the table template and if necessary can be quickly shown again.



Connected to table: Gallery_Media

Cell height (in rows): 5

☑ Allow Copy dataset

Display in overview	Variable	Type	Language dependent	Column width	DE	EN
☑	cs_picture	PICTURE		120	Media_Ref	Media_Ref
□	cs_name	TEXT	✓	180	Name_DE	Name_EN
☑	cs_description	TEXTAREA	✓	220	Description_DE	Description_EN

Figure 5-3: Configuration using the table template ("Mapping")

Multi-line display of datasets and all other presented configuration options via table template mapping are <u>*not*</u> *supported in WebEdit (see Chapter 4.3 page 74).*

5.1.4 Media gallery

Enhancement of the CMS_INPUT_ CONTENTLIST input component enables several media to be selected from the project's Media Store and to be arranged and displayed within a gallery view (see Chapter 4.1.6 page 35).

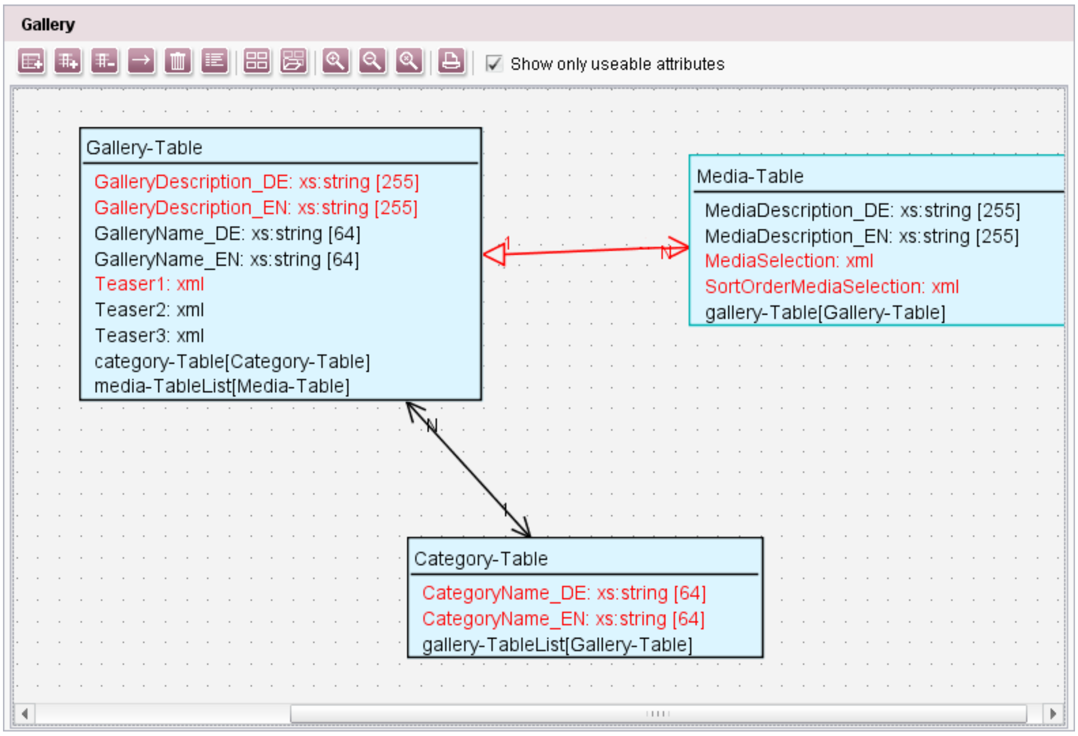
Two forms of display are possible:

Single galleries: These mostly consist of a homepage in which general information about the gallery, such as its name and description, are displayed and one or several teaser pictures, which are usually gallery pictures displayed with reduced size. The gallery's name, description and teaser pictures are linked. To this end, a reference to the medium from the Media Store is rendered within the template for each picture in the gallery, mostly with a larger resolution. The gathering together of media to form a group can be provided via an external JavaScript library, as used, for example, in the Mithras demo project (see Figure 4-23). Here all the pictures are grouped together in a gallery using an attribute (`rel`) within the template. This enables the display of a picture navigation ("picture 2 of 3") within the enlarged gallery view and paging through all the pictures in a gallery.

Gallery overviews: If there are several galleries in a project, these can be output filtered by categories (e.g. sport, politics, science) by defining queries.

The pictures to be used in the galleries must first (like all other media too) exist in the Media Store. All other data such as descriptive texts of the individual media and names, descriptive texts, etc. of the galleries, on the other hand, are saved and managed in data sources. To this end, two tables must be created, which the editor can subsequently maintain in the Content Store:

- Gallery table: In this table, the media of the individual galleries are assigned and general information on the galleries is managed.
- Media table: Information on the media which is to be displayed in the gallery is managed in this table.
- A categories table can be optionally created. It can be used for subsequent categorisation of the galleries.



Gallery

☑ Show only useable attributes

Gallery-Table
GalleryDescription_DE: xs:string [255]
GalleryDescription_EN: xs:string [255]
GalleryName_DE: xs:string [64]
GalleryName_EN: xs:string [64]
Teaser1: xml
Teaser2: xml
Teaser3: xml
category-Table[Category-Table]
media-TableList[Media-Table]

Media-Table
MediaDescription_DE: xs:string [255]
MediaDescription_EN: xs:string [255]
MediaSelection: xml
SortOrderMediaSelection: xml
gallery-Table[Gallery-Table]

Category-Table
CategoryName_DE: xs:string [64]
CategoryName_EN: xs:string [64]
gallery-TableList[Gallery-Table]

Figure 5-4: DB schema of a media gallery (example)

Because general data on the galleries and the media of the galleries is maintained in separate tables, media can be used more than once (in several galleries), each with different descriptive texts. At the same time, a medium in the media table is always only a reference to the respective object in the Media Store.

The required table templates must now be created below the database schema with the input components for the table columns created in the database schema. The **Gallery-Table** must contain the enhanced input component CMS_INPUT_CONTENTLIST for the selection of media from the media table (see Figure 4-24). This requires details of the `MEDIAMODE` tag. This tag is used to control object selection via the "Add" icon of the input component. At the same time, the unique identifier of the CMS_INPUT_PICTURE and/or CMS_INPUT_FILE input component of the media table must be given for the `mediaEditor` attribute. In addition, the `indexAttribute` attribute is used to store the order defined by the editor on selecting the media (see Figure 4-25). Optionally, other input components can be added to the template, for example, to maintain descriptions.

The picture input component for maintenance of the media of a gallery is defined in the **Media-Table**. The navigation mechanism for paging through the gallery is defined here using the PageGroup function (in the HTML channel).

These templates are already sufficient for the media gallery function and can be output in the project via content projections

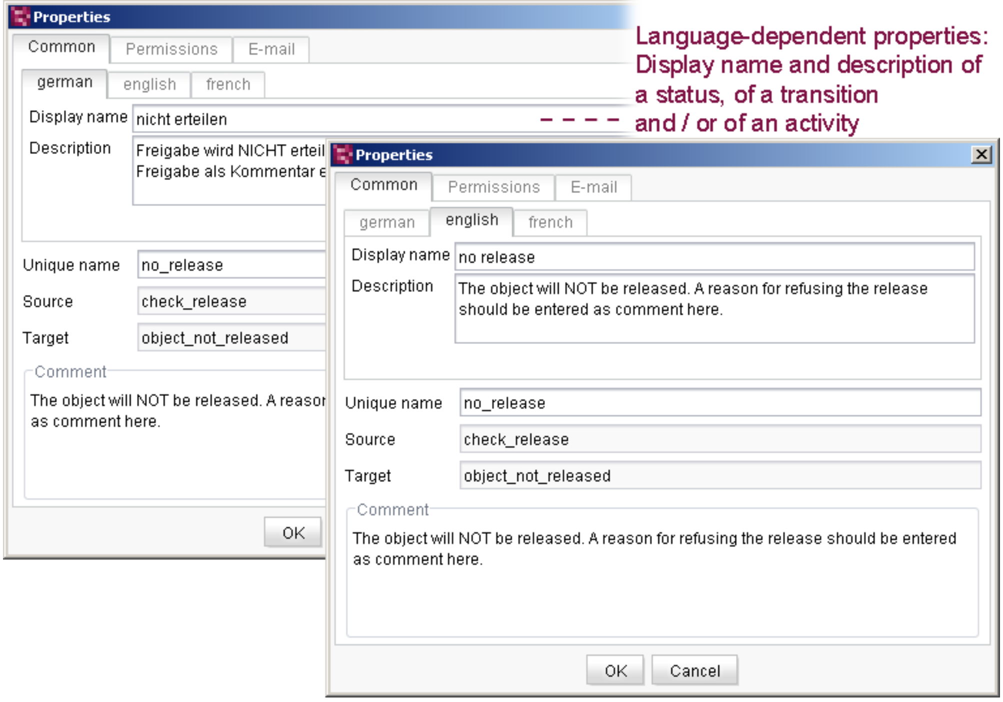
If the media gallery function is also to be made available to editors, a corresponding section template must be created. This is done using the CMS_INPUT_OBJECTCHOOSER input component, via which the datasets from the gallery table can be accessed. The input component can be configured so that only existing entries can be selected from the gallery table or new datasets can be created in the gallery table.

For further information on the media gallery, see FirstSpirit Online Documentation[1]*.*

The "media gallery" function is <u>*not*</u> *supported in WebEdit (see Chapter 4.3 page 74).*

5.1.5 Enhancement of workflow modelling

Workflow modelling enhancements were implemented in FirstSpirit Version 4.2. The status, activities and transitions properties can now be assigned language-dependent display names and descriptions.



Properties

Common | Permissions | E-mail

german | english | french

Display name: nicht erteilen

Description: Freigabe wird NICHT erteil
Freigabe als Kommentar e

Unique name: no_release

Source: check_release

Target: object_not_released

Comment: The object will NOT be released. A reasor
as comment here.

OK

Properties

Common | Permissions | E-mail

german | english | french

Display name: no release

Description: The object will NOT be released. A reason for refusing the release should be entered as comment here.

Unique name: no_release

Source: check_release

Target: object_not_released

Comment: The object will NOT be released. A reason for refusing the release should be entered as comment here.

OK | Cancel

Language-dependent properties:
Display name and description of
a status, of a transition
and / or of an activity

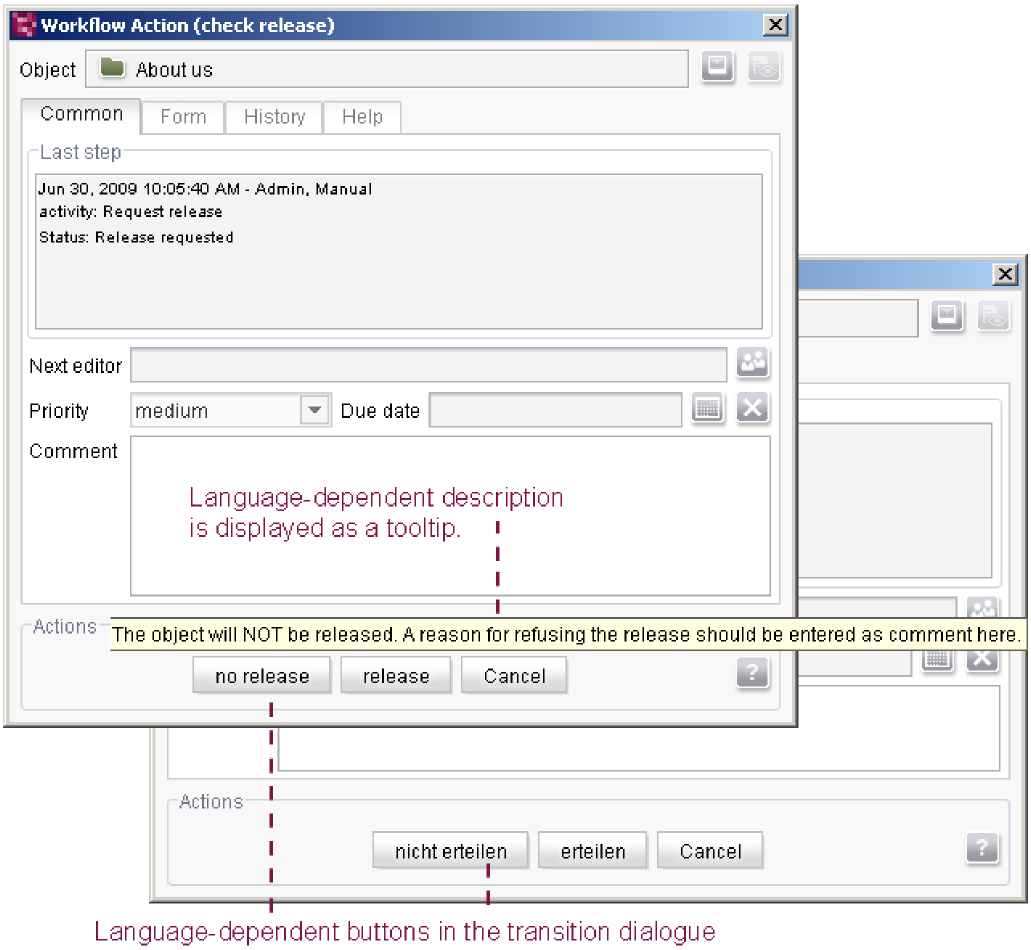
Figure 5-5: Language-dependent display names and comments for workflows

These are the *editing languages* (not the project languages). Editing languages are defined for a project by the project administrator and can then be configured by the

[1] .../ Advanced topics/Media galleries

editor using the "Extras – Preferred Display Language" menu. The editing languages affect, for example, language-dependent content which has been defined by the template developer, e.g. within the page or section templates. The relevant language-dependent labels are displayed to the editor, for example, in the form area (labelling of the input fields, tooltips, elements of a combobox, etc.).

In FirstSpirit Version 4.2, it is now possible for the first time ever to also provide the editor with language-dependent context menus and language-dependent buttons in the transition dialog, for connecting a workflow.



Workflow Action (check release)

Object: About us

Last step:

Jun 30, 2009 10:05:40 AM - Admin, Manual
activity: Request release
Status: Release requested

Priority: medium

The object will NOT be released. A reason for refusing the release should be entered as comment here.

Language-dependent description is displayed as a tooltip.

Language-dependent buttons in the transition dialogue

Figure 5-6: Language-dependent buttons in the Action dialog of a workflow

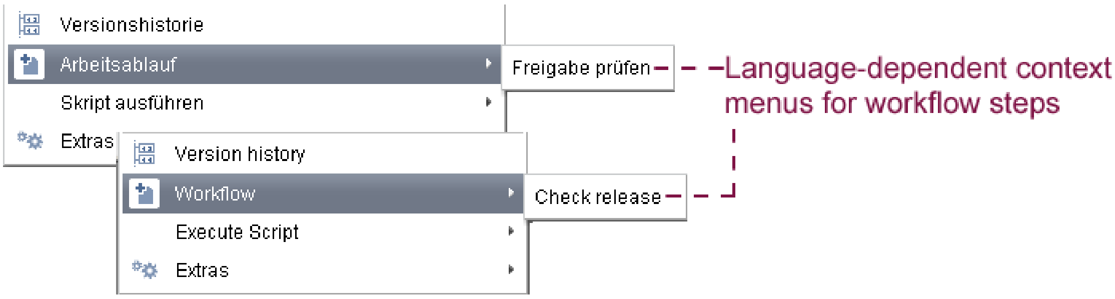


Figure 5-7: Language-dependent context menus on connecting a workflow

Note on support in WebEdit: In WebEdit, unlike in JavaClient, there are no editing languages. Here, language-dependent labels (for example in the form area) are displayed in the so-called GUI language ("locale"). If a language-dependent extension for a GUI language in WebEdit does not exist in the workflow, the "unique name" is displayed in WebClient instead.

5.1.6 WebEdit format templates for the "Easy-Edit" function

WebEdit 4.2 has been enhanced to include a new "Easy-Edit" function (see Chapter 4.2.2 page 68). The "Easy-Edit" function has been introduced to enable direct editing of sections within the preview page without using separate windows. It can therefore replace the Quick Edit function at section level.

Existing FirstSpirit projects do not have to be migrated. Easy-Edit is an additional function, i.e. existing projects can initially continue to be used as usual without adjustments (and therefore without Easy-Edit).

To use the "Easy-Edit" function, the templates of a project must be adjusted first. Special templates, the Easy-Edit format templates, are used for this. They can be combined with the WebEdit format templates and, just like these, are already included in the FirstSpirit scope of supply. The Easy-Edit format templates are created in the project together with the WebEdit format templates using the FirstSpirit server and project configuration, if the project property "Use WebEdit" is enabled. Apart from the format templates, a new media folder "WebClient Media (EasyEdit)" is created in the project for the Easy-Edit function.

Note: For release projects, the "WebClient Media (EasyEdit)" folder and the media contained in it must be manually released in the project. On disabling the project property "Use WebEdit", the format templates and the "WebClient Media (EasyEdit)" media folder are removed from the project. In this case, release on the higher level folder of the Media Store takes place automatically.

The Easy-Edit format templates must be added to the page, section and/or table templates in which the Easy-Edit function is to be used.

A precise description of the format templates is given in the FirstSpirit Online Documentation[2].

Note: The technologies used for the "Easy-Edit" function intervene in the HTML source code of a FirstSpirit project far more than the WebEdit functions used to date. For this reason, it is not possible to guarantee that "Easy-Edit" can be used without changes to the project HTML. Use of JavaScript must at least be enabled within the browser settings; however, other changes to the browser configuration through to adjustments in the project may be necessary.
Furthermore, problems can occur if JavaScript Frameworks are used in projects, as Easy-Edit uses the JavaScript Framework MooTools. Manual adjustments may also be necessary.

Note: The "Easy-Edit" function already provides new user prompting concepts in WebEdit 4.2, which will be continued with the launch of WebEdit 5.0. However, in WebEdit 5.0, the technologies on which these are based will be implemented in a completely new way with the help of the GWT framework[3]. In order to keep the migration work for projects as small as possible, it is planned to keep template changes to the Easy-Edit function of Version 4.2 compatible with Version 5.0. However, this cannot be guaranteed at the present time.

5.1.7 Format templates for the "Content Highlighting" function

The "Content Highlighting" function was introduced to make it easier for editors to search for content and to navigate in JavaClient (see Chapter 4.1.5 page 30).

Existing FirstSpirit projects do not have to be migrated. Content Highlighting is an additional function, i.e. existing projects can initially continue to be used as usual without adjustments (and therefore without Content Highlighting).

Analogous to use of Easy-Edit, the templates of a project must be adjusted first for use of the “Content Highlighting" function. Special format templates and a CSS stylesheet are used for this; these are already included in the FirstSpirit scope of supply. The format templates must be added to the page, section and/or table templates in which the Content Highlighting function is to be used.

[2] FirstSpirit Online Documentation - Chapter: ../Template development/WebEdit/Easy-Edit

[3] GWT – Google Web Toolkit

A precise description of the format templates and of the stylesheet is given in the FirstSpirit Online Documentation[4].

Note: The technologies used for the "Content Highlighting" function intervene in the HTML source code of a FirstSpirit project far more than the functions used to date. As the JavaScript share here is smaller than, for example, when using "Easy-Edit", conflicts should occur far less often. Nevertheless, it is not possible to guarantee that "Content Highlighting" can be used without changes to the project HTML. Particularly "pixel-precise" layouts in conjunction with CHTML can cause problems here, as several additional pixels are required in the HTML environment due to the framing of the highlighted content.

5.1.8 Templates: Simplified syntax for lambda expressions

The template language for lambda expressions has been simplified for FirstSpirit Version 4.2.

Example of use of lambda expression:

```
$CMS_SET(immos, [
{"Name":"A", "Address":{"City":"Dortmund", "ZIP":42400,
"Street":"Amsterdamer Platz", "No":"32"}, "Price":177000,
"sqm":190},
{"Name":"B", "Address":{"City":"Bochum",   "ZIP":44877,
"Street":"Allestr.", "No":"114"}, "Preis":150000, "sqm":150},
{"Name":"C", "Address":{"City":"Hagen",    "ZIP":42345,
"Street":"Kleiner Weg", "No":"2b"}, "Price":555000,
"sqm":320},
{"Name":"D", "Address":{"City":"Bochum",   "ZIP":44890,
"Street":"Zeppelindamm", "No":"93"}, "Price":352000,
"sqm":210}])$
```

Example (old notation from FirstSpirit Version 4.1):

Most expensive property:

```
    $CMS_VALUE(immos.max(lambda(x : x.Price)))$<br>
```

Example (new notation in FirstSpirit Version 4.2):

Most expensive property:

```
    $CMS_VALUE(immos.max(x -> x.Price))$<br>
```

[4] FirstSpirit Online Documentation - Chapter: ../Advanced topics/Content Highlighting

5.1.9 Templates: New data element CMS_INCLUDE_OPTIONS

In FirstSpirit there are several input components which offer a set of values for being selected by the editor (e.g. CMS_INPUT_CHECKBOX or CMS_INPUT_ COMBOBOX). Using the new data element *CMS_INCLUDE_OPTIONS* this set of values can be configured dynamically. This means that the input component can be filled automatically, for example, with data sets from an external table, project languages, output channels etc.

Note: *CMS_INCLUDE_OPTIONS* expands the options of the data element *CMS_INCLUDE_CONTENT*. In FirstSpirit version 4.2 *CMS_INCLUDE_CONTENT* can be used parallel with the new data element *CMS_INCLUDE_OPTIONS*. From FirstSpirit version 5.0 the data element *CMS_INCLUDE_CONTENT* will no more be supported.

The following example compares the use of *CMS_INCLUDE_OPTIONS* and *CMS_INCLUDE_CONTENT*.

A combobox from which news from a data source (in this example: "Products.press_releases") can be selected language-dependent by the editor on the basis of the headline (in this example column “Headline") can be created as follows using *CMS_INCLUDE_CONTENT*:

```
<CMS_INPUT_COMBOBOX name="cs_pressReleases" useLanguages="no">
   <CMS_INCLUDE_CONTENT table="Products.press_releases"
type="entries">
      <LABEL lang="*">#row.Headline</LABEL>
   </CMS_INCLUDE_CONTENT>
   <LANGINFOS>
      <LANGINFO lang="DE" label="Pressemitteilung"/>
      <LANGINFO lang="*" label="Press release"/>
   </LANGINFOS>
 </CMS_INPUT_COMBOBOX>
```

The according syntax for *CMS_INCLUDE_OPTIONS* is:

```
<CMS_INPUT_COMBOBOX name="cs_pressReleases" useLanguages="no">
    <CMS_INCLUDE_OPTIONS type="DATABASE">
      <LABELS>
        <LABEL lang="*">#item.Headline</LABEL>
      </LABELS>
      <TABLE>Products.press_releases</TABLE>
    </CMS_INCLUDE_OPTIONS>
    <LANGINFOS>
      <LANGINFO lang="DE" label="Pressemitteilung"/>
      <LANGINFO lang="*" label="Press release"/>
    </LANGINFOS>
  </CMS_INPUT_COMBOBOX>
```

5.1.10 Templates: Output of the menu name using the function MenuGroup

From FirstSpirit version 4.2 the method *#nav.label* returns the value of the field „Menu name“ of the menu of the respective menu level (from the Site Store) when using the function *MenuGroup*.

Up to FirstSpirit version 4.1 inclusivley, method *#nav.label* returned the reference name of the start page of the menu level.

Example for the use of the *MenuGroup* function with *#nav.label*:

```
<CMS_FUNCTION name="MenuGroup" resultname="fr_pt_menuGroup">

<CMS_CDATA_PARAM name="firstAvailable">
          <![CDATA[$CMS_VALUE(#nav.label)$]]></CMS_CDATA_PARAM>
<CMS_CDATA_PARAM name="firstNotAvailable">
          <![CDATA[---]]></CMS_CDATA_PARAM>
<CMS_CDATA_PARAM name="lastAvailable">
          <![CDATA[$CMS_VALUE(#nav.label)$]]></CMS_CDATA_PARAM>
<CMS_CDATA_PARAM name="lastNotAvailable">
          <![CDATA[---]]></CMS_CDATA_PARAM>
<CMS_CDATA_PARAM name="previousAvailable">
          <![CDATA[$CMS_VALUE(#nav.label)$]]></CMS_CDATA_PARAM>
<CMS_CDATA_PARAM name="previousNotAvailable">
          <![CDATA[---]]></CMS_CDATA_PARAM>
<CMS_CDATA_PARAM name="nextAvailable">
          <![CDATA[$CMS_VALUE(#nav.label)$]]></CMS_CDATA_PARAM>
<CMS_CDATA_PARAM name="nextNotAvailable">
          <![CDATA[---]]></CMS_CDATA_PARAM>
<CMS_PARAM name="cycle" value="1" />

</CMS_FUNCTION>
```

5.1.11 New input components (Status: Under development)

Change to the input component model: FirstSpirit Version 4.2 marks the start of fundamental revision and consolidation of FirstSpirit's input component model (cf. "FirstSpirit Roadmap 2009-2012"). Within the scope of these activities, a whole range of input components previously implemented separately will be brought together. It is planned to bring together the following input component groups:

- Single value input components: Links to other FirstSpirit objects, e.g. CMS_INPUT_FILE, CMS_INPUT_PICTURE, CMS_INPUT_PAGEREF, etc.
- Set-valued input components: CMS_INPUT_CONTENTLIST, CMS_INPUT_TABLIST, CMS_INPUT_CONTENTAREALIST, CMS_INPUT_LINKLIST

This is an extensive consolidation project in which compatibility and migration aspects also play an important role. This is why the implementation is in two phases:

- With **FirstSpirit Version 4.2**, a new generation of input components will be gradually introduced with the name prefix "FS_" instead of "CMS_INPUT_". These new input components will be gradually added within the scope of continued development of FirstSpirit and will be adjusted to clients' needs.
- The new input components cannot be officially released until **FirstSpirit Version 5.0**. This process is necessary, as release of the components in Version 4.2 would already bindingly specify the parameter assignment and API. This would make flexible further development of the components, taking into account customers' interests, no longer possible.

Status – Under development: As the official release will not take place until FirstSpirit Version 5.0, the input components ("FS_") can already be used in FirstSpirit Version 4.2, but in Version 4.2 their status is "**under development**". In specific terms, this status means:

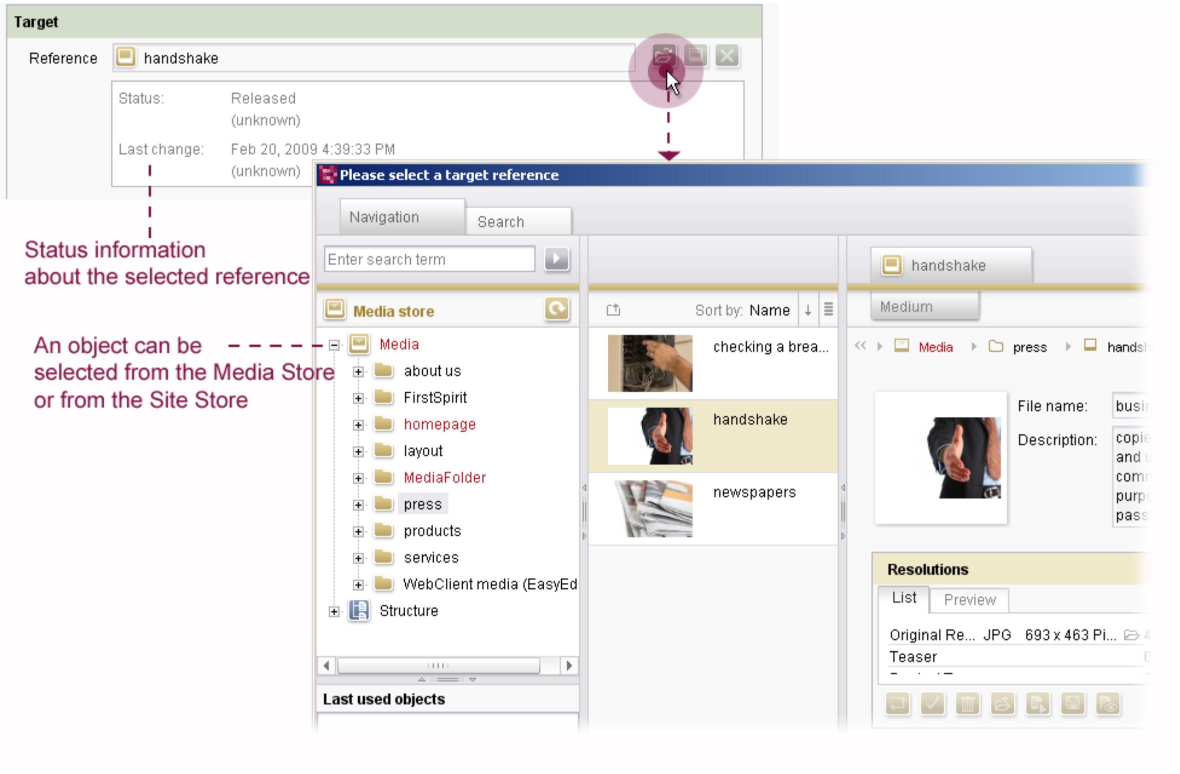
- The input component is supported within the scope of the usual quality assurance and debugging.
- The persistence format will be kept compatible during the continued development, i.e. once data has been entered, it can continue to be imported. However, there is no downwards compatibility with FirstSpirit Version 4.1.
- The aim is not to make too many changes to the user prompting. However, as especially optimisation of the user prompting, is the objective of the iterative procedure, editorially relevant changes may well occur
- The aim is to keep the parameter assignment of the input component compatible. Should this not be possible, a corresponding announcement will be made within the scope of the "FirstSpirit Release Notes".
- The API of the input components will change during the course of the

development. If possible (and meaningful) the API changes will be compatible. However, incompatible changes are possible and will also be announced within the scope of the "FirstSpirit Release Notes".

- As release of the "FS_" components is not planned until FirstSpirit Version 5.0 with WebEdit 5.0, support for these component in WebEdit 4.2 will be very rudimentary. An exception here is all new input components which are necessary for the migration to generic link editors.

The implications for use in productive projects are: Use of the components is possible, if the project developers dispense with use of the API and they are willing to potentially subsequently adjust the parameter assignment of the input components. The editorial users should be prepared for changes in the user prompting. Should this be unacceptable, use of the new input components should be dispensed with in Version 4.2.

FS_REFERENCE: The new input component FS_REFERENCE is for holding any reference and therefore includes the function of the input components: CMS_INPUT_PICTURE, CMS_INPUT_FILE and CMS_ INPUT_PAGEREF.



Target

Reference: handshake

Status: Released (unknown)

Last change: Feb 20, 2009 4:39:33 PM (unknown)

Status information about the selected reference

An object can be selected from the Media Store or from the Site Store

Please select a target reference

Figure 5-8: FS_REFERENCE input component with selection dialog

Unlike the conventional input components for referencing objects, for example, CMS_INPUT_PICTURE, here the type of referenced object is not defined directly via the input component. Restriction to specific types, e.g. only pictures or only files can, however, be made by the template developer by configuring the component.

The selection options for the editor can be limited to such an extent that only specific references can be selected (`<ALLOW type=.../>` or `<HIDE type=.../>`), selection of other references is prevented or a selection can only be made from certain Stores or folders (`<FOLDER name=... store=.../>`). For example, the component can be configured so that the editor can select pictures and files, but not page references.

```
      <FS_REFERENCE name="reference" ...>
         <FILTER>
           <ALLOW type="picture"/>
           <ALLOW type="file"/>
            <HIDE type="pageref"/>
         </FILTER>
         <LANGINFOS>
           ...
         </LANGINFOS>
         <PROJECTS>
           <LOCAL name=".">
             <SOURCES>
               <FOLDER name="root" store="mediastore"/>
             </SOURCES>
           </LOCAL>
         </PROJECTS>
       </FS_REFERENCE>
```

A component configured in this way can therefore reproduce the "mediaref“ field an "internalLink“ used to date.

Support in WebEdit: The input component is also supported in FirstSpirit WebClient. The following restrictions currently apply:

- A selection option exists in WebEdit for the following Stores only: Page Store, Media Store, Content Store and Site Store.
- Uploading new objects is not supported in WebEdit.
- Document groups can not be selected in WebEdit.

For further information on the FS_REFERENCE input component, see FirstSpirit Online Documentation[5].

FS_DATASET: Analogous to the CMS_INPUT_OBJECTCHOOSER input component, the new input component FS_DATASET is used to reference a dataset from the Content Store (cf. Chapter 4.1.12 page 54).

Unlike the selection option within the input component CMS_INPUT_OBJECTCHOOSER, FS_DATASET is not limited by the `table`

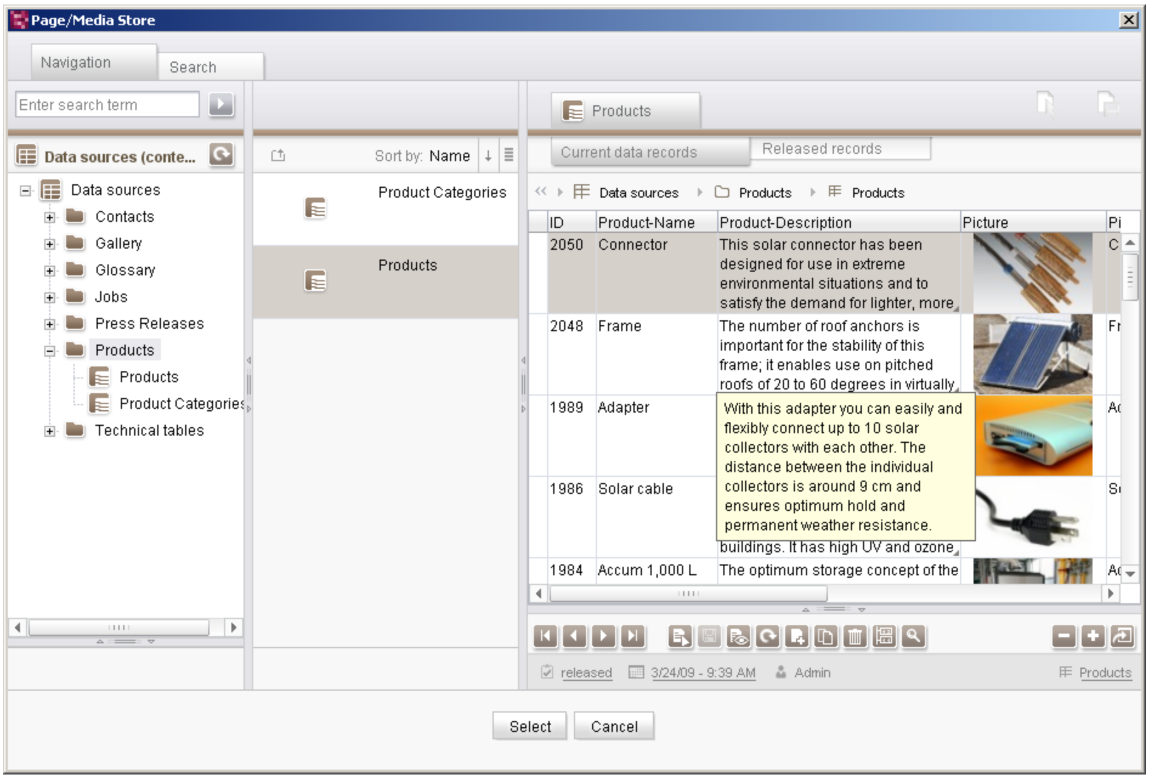
[5] ../Template development/Forms/Input components (new)/FS_REFERENCE

parameter to selection of datasets from a defined target table. Instead, the required table can be selected using a selection dialog. At the same time, the target table and required dataset are selected (see Figure 5-9).

The selection options for the editor can be limited to such an extent that only specific tables from the Content Store can be selected (`<CONTENT name=.../>`; selection of other tables is prevented.

The new input component will also provide all editing options which are also possible for the input component CMS_INPUT_OBJECTCHOOSER (cf. Chapter 4.1.12 page 54)

Note: The standard process for deleting datasets has been changed for the FS_DATASET input component (compared to CMS_INPUT_ OBJECTCHOOSER). Deleting is allowed as a default in FS_DATASET (Parameter `allowDelete=YES`).



ID	Product-Name	Product-Description	Picture	Pi
2050	Connector	This solar connector has been designed for use in extreme environmental situations and to satisfy the demand for lighter, more		C
2048	Frame	The number of roof anchors is important for the stability of this frame; it enables use on pitched roofs of 20 to 60 degrees in virtually		Fr
1989	Adapter	With this adapter you can easily and flexibly connect up to 10 solar collectors with each other. The distance between the individual collectors is around 9 cm and ensures optimum hold and permanent weather resistance.		Ad
1986	Solar cable	buildings. It has high UV and ozone		S
1984	Accum 1,000 L	The optimum storage concept of the		Ac

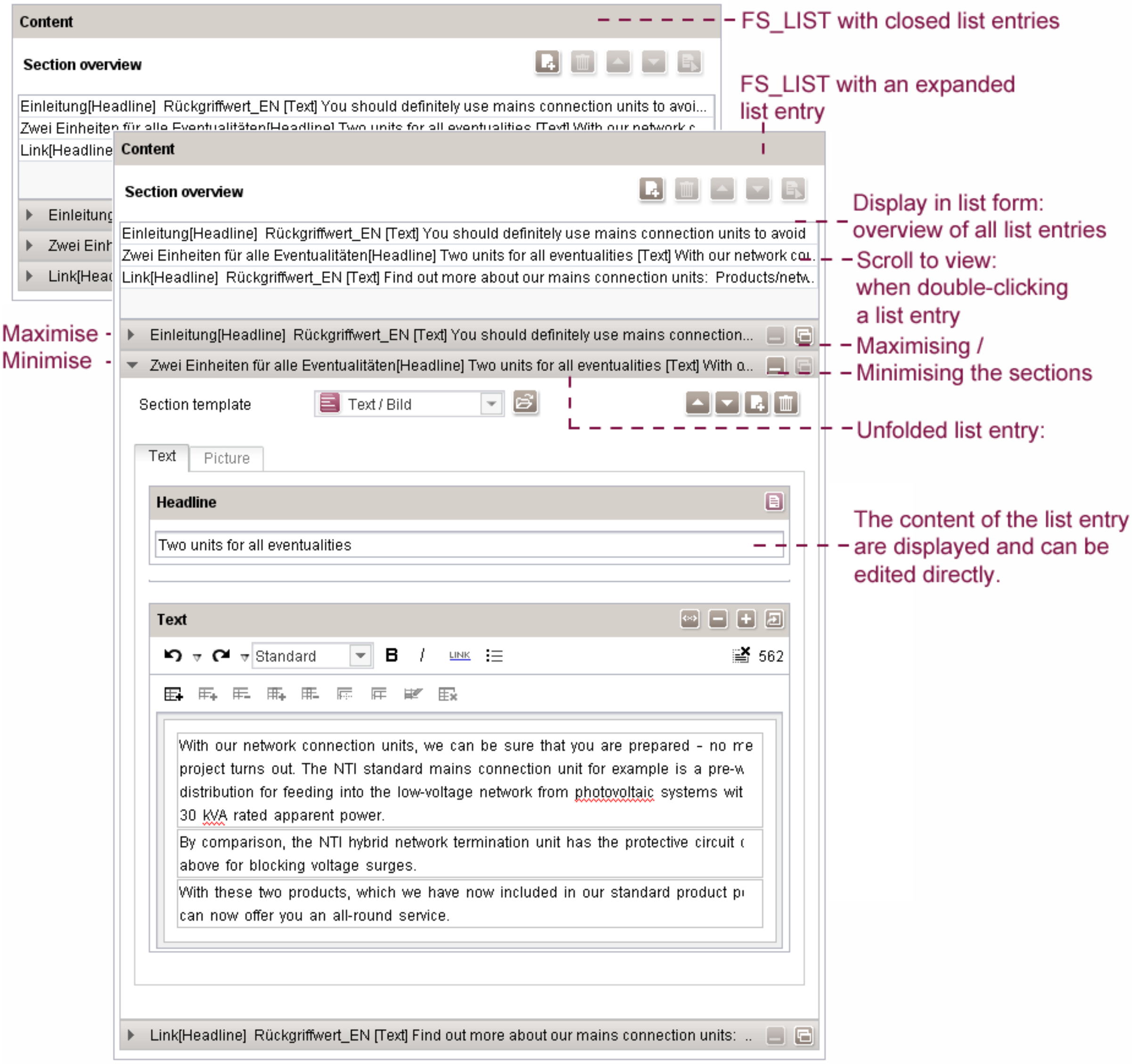
Figure 5-9: Selection dialog for target tables and datasets

Support in WebEdit: The input component is also supported in FirstSpirit WebClient – albeit with restrictions:

- Hierarchical structures of the Content Store cannot be displayed in WebEdit, if the selection options have been limited using the <CONTENT...> tag. In this case, the allowed tables in WebEdit are displayed directly below the root node.
- For details of further restrictions, cf. Chapter 4.2.4 page 71.

For further information on the FS_DATASET input component, see FirstSpirit Online Documentation[6].

FS_LIST: The new FS_LIST input component enables several sections to be selected and displayed as a list. It can therefore replace the familiar input component CMS_INPUT_CONTENTAREALIST, whereby FS_LIST can do completely without pop-up dialogs.



Einleitung[Headline] Rückgriffwert_EN [Text] You should definitely use mains connection units to avoid
Zwei Einheiten für alle Eventualitäten[Headline] Two units for all eventualities [Text] With our network co…
Link[Headline] Rückgriffwert_EN [Text] Find out more about our mains connection units: Products/netw…

Einleitung[Headline] Rückgriffwert_EN [Text] You should definitely use mains connection...
Zwei Einheiten für alle Eventualitäten[Headline] Two units for all eventualities [Text] With o..

Two units for all eventualities

With our network connection units, we can be sure that you are prepared - no me project turns out. The NTI standard mains connection unit for example is a pre-w distribution for feeding into the low-voltage network from photovoltaic systems wit 30 KVA rated apparent power.

By comparison, the NTI hybrid network termination unit has the protective circuit (above for blocking voltage surges.

With these two products, which we have now included in our standard product p can now offer you an all-round service.

Link[Headline] Rückgriffwert_EN [Text] Find out more about our mains connection units: ..

FS_LIST with closed list entries

FS_LIST with an expanded list entry

Display in list form: overview of all list entries

Scroll to view: when double-clicking a list entry

Maximising / Minimising the sections

Maximise
Minimise

Unfolded list entry:

The content of the list entry are displayed and can be edited directly.

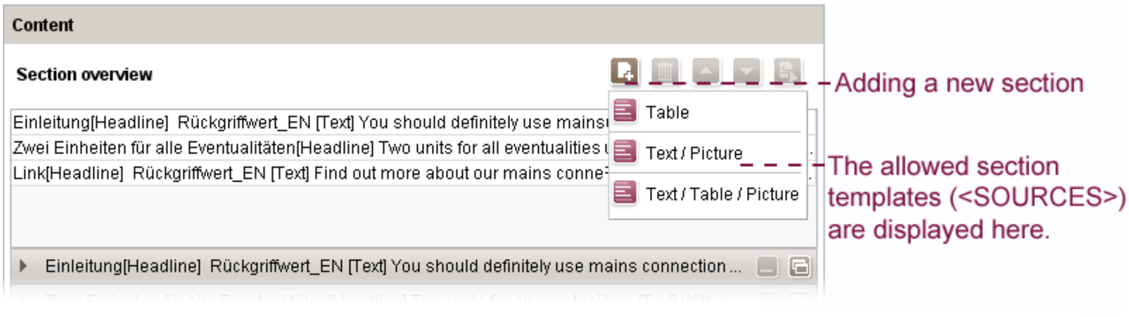
Figure 5-10: FS_LIST input component with direct editing option

All contents of the list entries in FS_LIST are directly visible and editable. The original list display is used as a type of table of contents. When the list entry is selected, the actual content of the referenced sections is shown below the list display

[6] ../Template development/Forms/Input components (new)/FS_DATASET

and can also be directly edited there. In addition, the editor can double click the required list entry to switch directly to the contents of the entry ("Scroll-to-View"). The individual sections below the list display can also be maximised or minimised using the icons or by double-clicking the window frame (see Figure 5-10).

On adding a new entry, the section template can now also be directly selected (without a pop-up window opening):



Einleitung[Headline] Rückgriffwert_EN [Text] You should definitely use mains
Zwei Einheiten für alle Eventualitäten[Headline] Two units for all eventualities
Link[Headline] Rückgriffwert_EN [Text] Find out more about our mains conne
Einleitung[Headline] Rückgriffwert_EN [Text] You should definitely use mains connection ...

Adding a new section

The allowed section templates (<SOURCES>) are displayed here.

Figure 5-11: Addition of a new section (selection of a section template)

```
<FS_LIST name="fs_list" hFill="yes" rows="10">
    <LANGINFOS>
      ...
    </LANGINFOS>
    <SOURCES>
      <TEMPLATE name="textpicture"/>
      <TEMPLATE name="textpicture_1"/>
      <TEMPLATE name="table"/>
    </SOURCES>
  </FS_LIST>
```

Further development of FS_LIST should provide an additional option in FirstSpirit Version 5.0 for selecting datasets from the Content Store (to date CMS_INPUT_CONTENTLIST and CMS_INPUT_TABLIST) and for selecting links (to date CMS_INPUT_LINKLIST).

Support in WebEdit: The new input component FS_LIST is also supported in WebEdit – with restrictions but there, for the time being, it does not provide any options for direct editing of content. The functional scope of the FS_LIST input component corresponds to the CMS_INPUT_CONTENTAREALIST input component in WebEdit.

For further information on the FS_LIST input component, see FirstSpirit Online Documentation[7]*.*

[7] ../ Template development/Forms/Input components (new)/FS_LIST

5.1.12 Definition of content areas via the template properties

GUI support for the **definition of content areas** in the Template Store is available from FirstSpirit Version 4.2 on the "Properties" tab of a page template. Click the "Add Content Area" icon to add a new content area to a page template, edit, resort or delete an existing content area (see Figure 5-12).

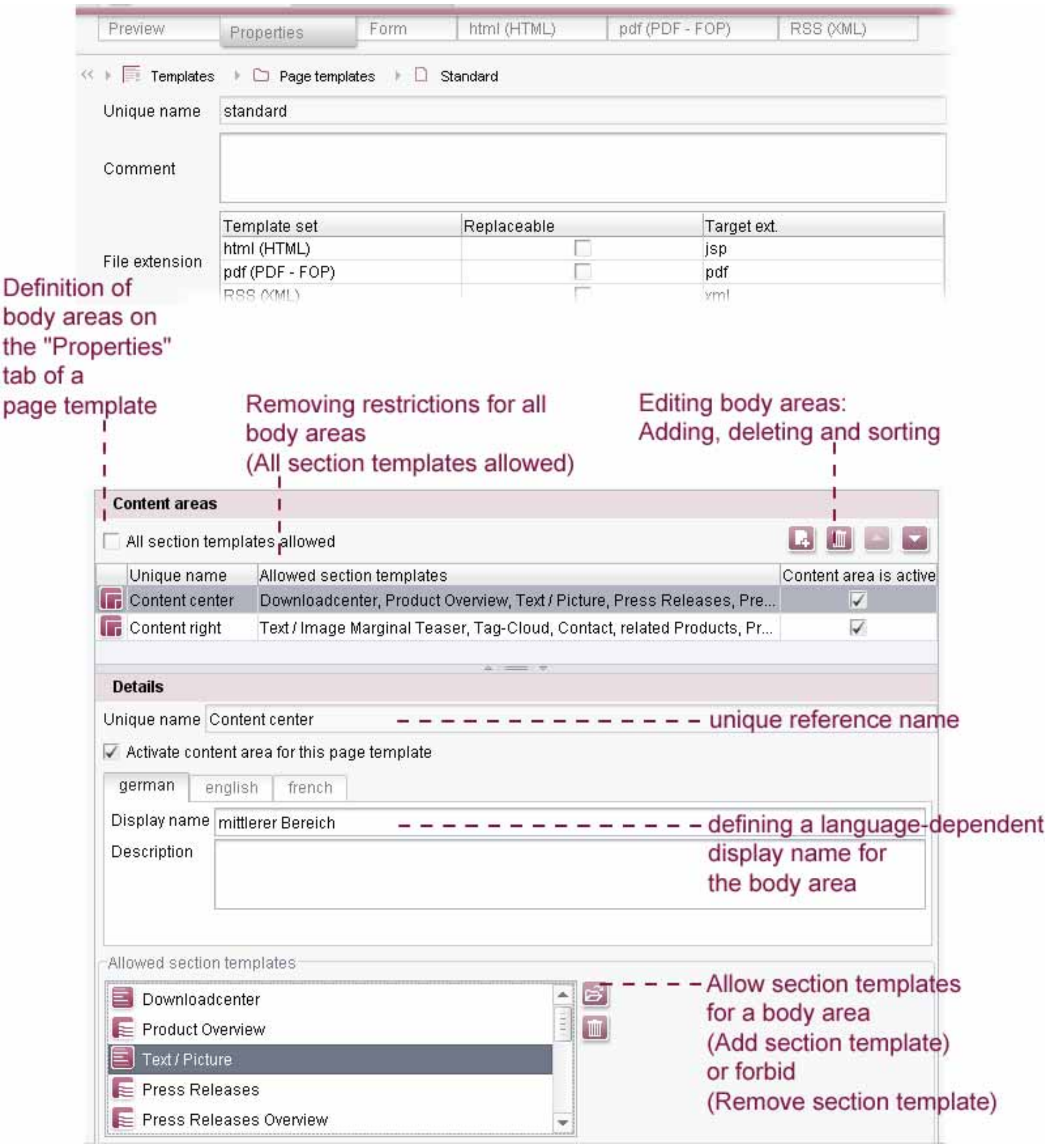
In addition, the "Properties" tab can also be used to define the **section restrictions** for the page template. To this end, the required section templates can either be allowed or prohibited by adding them to or removing them from a list (for a content area). For the corresponding content areas, this means that only the respective selected section templates are allowed. The definition of section restrictions used to date by using Drag & Drop to drop section templates onto a content area, is therefore no longer supported.

Optionally, all section templates can also be allowed for all content areas of a page template. In this way, each section restriction for the page template is cancelled.

The addition of **language-dependent display names** for content areas is also new (see Figure 5-12). Content areas can now be assigned one (or several) language-dependent display names and a unique reference name. If a "preferred display language" is defined in FirstSpirit JavaClient, the corresponding language-dependent display names of the content areas are displayed in JavaClient.

Note: The language-dependent display names for content areas are not supported in WebEdit (see Chapter 4.3 page 74).

For details of automatic adjustment of templates of projects from FirstSpirit Version 4.1 to FirstSpirit Version 4.2, see Chapter 8.1.5 (from page 115).



Unique name: standard

Comment:

File extension:

Template set	Replaceable	Target ext.
html (HTML)	□	jsp
pdf (PDF - FOP)	□	pdf
RSS (XML)	□	xml

Definition of body areas on the "Properties" tab of a page template

Removing restrictions for all body areas (All section templates allowed)

Editing body areas: Adding, deleting and sorting

Content areas

□ All section templates allowed

Unique name	Allowed section templates	Content area is active
Content center	Downloadcenter, Product Overview, Text / Picture, Press Releases, Pre...	☑
Content right	Text / Image Marginal Teaser, Tag-Cloud, Contact, related Products, Pr...	☑

Details

Unique name: Content center – unique reference name

☑ Activate content area for this page template

Display name: mittlerer Bereich – defining a language-dependent display name for the body area

Description:

Allowed section templates

- Downloadcenter
- Product Overview
- Text / Picture
- Press Releases
- Press Releases Overview

Allow section templates for a body area (Add section template) or forbid (Remove section template)

Figure 5-12: Defining content areas for a page template

5.1.13 Editing and deleting preview images

Preview images can now also be edited and deleted in the tab „Preview" of a page or section template using the context menu „Extras – Select preview image" and "Extras – Remove preview image". These items are only active if the respective template is in the edit mode.

5.2 Enhancements in FirstSpirit modules

5.2.1 "FirstSpirit Office" module (Word import)

The "FirstSpirit Office" module provides extensive infrastructure for importing MS Word documents into FirstSpirit input components. The basic data acceptance requirements in the DOM Editor (CMS_INPUT_DOM) and DOM table (CMS_INPUT_DOMTABLE) input components are initially covered with the introduction of the module in FirstSpirit Version 4.2. Further functions will then be implemented if necessary in subsequent FirstSpirit versions.

The basic functions of the module, implemented with FirstSpirit Version 4.2, can be used to import selected text passages (or whole documents from MS Word into a FirstSpirit input component using the clipboard (by means of "Copy and Paste").

The following content should be copied with the data acceptance:

- Texts
- Headings (h1 to h4)
- Text formatting for bold, italic and underline
- Paragraph acceptance including recognition of paragraph end (irrespective of Word format) and transfer to a FirstSpirit section format
- SHIFT RETURN (line end without section end)
- Lists (without formatting)
- Pictures/images (including automatic upload into the project's Media Store)
- Links (based on the default link templates
- Simple tables with copying of merging, cell colour, cell alignment and other formatting

After installing the module, the FirstSpirit server and project configuration can be used to define rules for the conversion of CF HTML tags into FirstSpirit expressions. This definition can be made globally (server-wide) and if necessary can be extended to include project-specific rulesets for individual projects.

Further functions will then probably be implemented in subsequent FirstSpirit versions. These include:

- Copying of complex formatting: This would include, for example, reproduction of the formatting on FirstSpirit format templates (character and section format templates), reproduction of individual formatting styles from Word (e.g. fonts / size / colour) on format template variables in FirstSpirit, copying of list and table styles,

- page and section generation and
- importing of files (via Office HTML).

The "FirstSpirit Office" function is <u>not</u> supported in WebEdit (see Chapter 4.3 page 74).

5.2.2 FirstSpirit RemoteMedia

The licence-dependent FirstSpirit remote media function has been enhanced for the input components CMS_INPUT_FILE, CMS_INPUT_PICTURE and CMS_INPUT_PAGEREF. Unlike FirstSpirit Version 4.1, in Version 4.2 it is now possible to define several remote projects (previously precisely one) for an input component.

To this end, a new `<PROJECTS>` tag was introduced for the configuration of input components within the form area. Within this project definition, it is possible to specify the following tags:

- LOCAL: This tag must be set once for each project definition and allows the definition of an upload folder in the local target project.
  <u>Parameters:</u>
  `name` default configuration for local projects `<LOCAL name="."/>.` The `name` parameter may not be empty.
  `uploadFolder` details of an upload folder for uploading files and media into the local project.
- REMOTE: This tag can have multiple definitions.
  <u>Parameters:</u>
  `name` Symbolic project name of the remote project.
  `uploadFolder` details of an upload folder for uploading files and media in the remote project.
  `autoReleaseAfterUpload` controls automatic release of media after they have been uploaded into the remote project.
- CATEGORY: This tag can have multiple definitions.
  <u>Parameters:</u>
  `name` Details of a category name.

<u>Note:</u> If two or more REMOTE or CATEGORY tags are defined with the same name, the first definition only is always followed.

<u>Note:</u> If a category already contains defined remote projects, these projects from the category are ignored.

The order of the definitions within the form area also affects the order in the selection dialog and upload dialog (the local project is excluded from this sorting).

Configuration example (new syntax):

```
<PROJECTS>
  <LOCAL name="." uploadFolder="local_folder"/>
  <REMOTE name="warehouse" uploadFolder="product_pictures"/>
  <CATEGORY name="remotemedia"/>
</PROJECTS>
```

The `remote`, `remoteOnly`, `uploadFolder` and `remoteUploadFolder` attributes used to date are therefore dropped in Version 4.2.

For details of automatic adjustment of the syntax of FirstSpirit Version 4.1 projects to FirstSpirit Version 4.2, see Chapter 8.1.3 (from page 114).

Note: These changes are not downwards compatible (see Chapter 7.2 page 110).

Multiple definition of remote projects is also supported in WebEdit, albeit with restrictions.

Support in WebEdit 4.2:

- The definition of several remote projects (`<REMOTE...>`) and specification of categories (`<CATEGORY...>`) is only supported in WebEdit for the CMS_INPUT_FILE and CMS_INPUT_ PICTURE input components.

Other restrictions:

- If several remote projects are configured for an input component, each with one `uploadFolder`, only the `uploadFolder` is taken into account in WebEdit. Further `uploadFolder` of other remote projects are ignored in WebEdit.

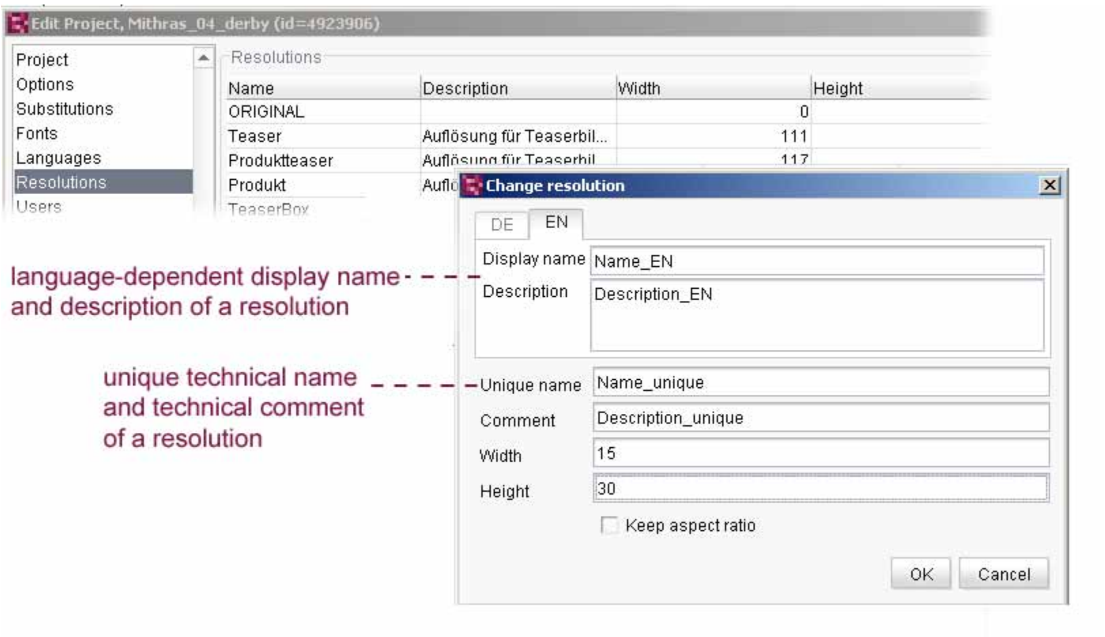
6 New Functions for Administrators

6.1 Server and project configuration enhancements

6.1.1 Extension of the language-dependent display options

Analogous to JavaClient, the language-dependent display options for projects have now been enhanced in the server and project configuration too.

Resolutions can now be assigned a language-dependent display name and a language-dependent description within the project properties:



Name	Description	Width	Height
ORIGINAL		0	
Teaser	Auflösung für Teaserbil...	111	
Produktteaser	Auflösung für Teaserbil	117	
Produkt	Auflö		
TeaserBox			

Display name: Name_EN
Description: Description_EN
Unique name: Name_unique
Comment: Description_unique
Width: 15
Height: 30
□ Keep aspect ratio

language-dependent display name and description of a resolution

unique technical name and technical comment of a resolution

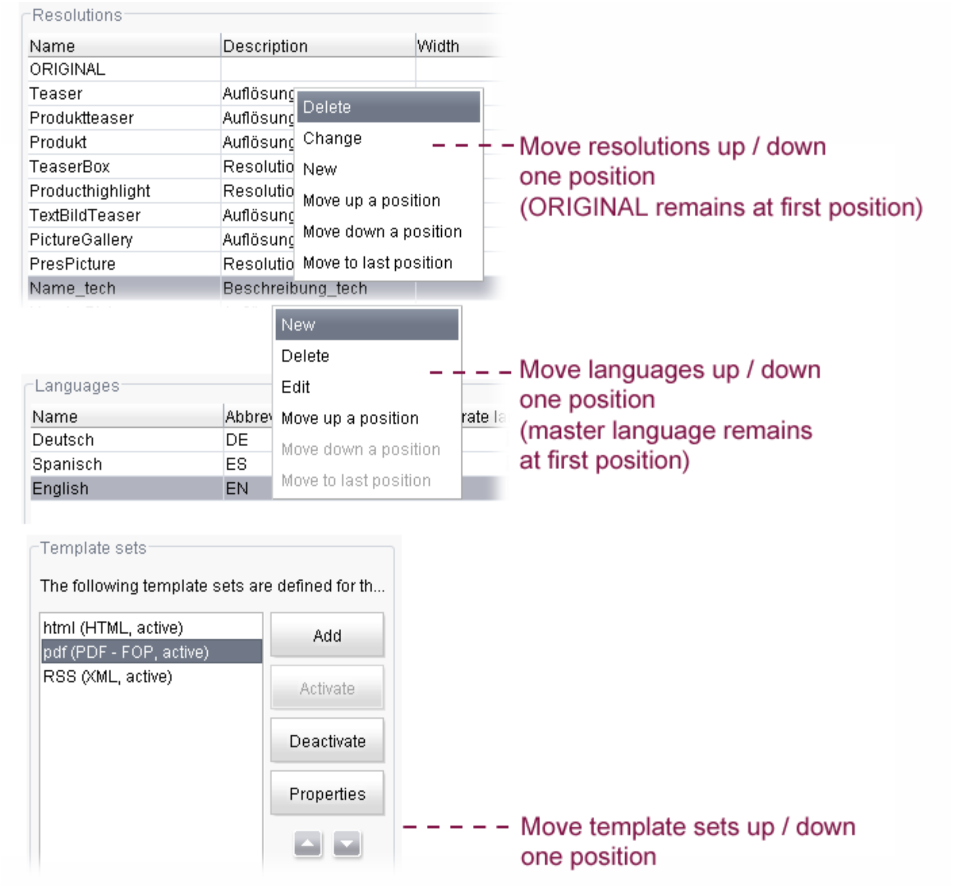
Figure 6-1: Assignment of a language-dependent display name for a resolution

Analogous to this, language-dependent display of **template sets** and **project languages** has also been added within the project properties and for **language templates** and the so-called **"Message of the day"** within the server properties.

If a "preferred display language" is defined in FirstSpirit JavaClient, the corresponding language-dependent display names of the resolutions, project languages and template sets and the language-dependent "Message of the day" are displayed in JavaClient.

6.1.2 Sort option for resolutions, languages, template sets

Apart from the addition of language-dependent inputs, template sets and language identifiers of a project can be resorted within the server and project configuration. To this end, the relevant entry simply has to be selected and gradually moved up or down using the context menu or the relevant buttons.



Resolutions

Name	Description	Width
ORIGINAL		
Teaser	Auflösung	
Produktteaser	Auflösung	
Produkt	Auflösung	
TeaserBox	Resolutio	
Producthighlight	Resolutio	
TextBildTeaser	Auflösung	
PictureGallery	Auflösung	
PresPicture	Resolutio	
Name_tech	Beschreibung_tech	

Move resolutions up / down one position (ORIGINAL remains at first position)

Languages

Name	Abbrev	rate la
Deutsch	DE	
Spanisch	ES	
English	EN	

Move languages up / down one position (master language remains at first position)

Template sets

The following template sets are defined for th...

- html (HTML, active)
- pdf (PDF - FOP, active)
- RSS (XML, active)

Move template sets up / down one position

Figure 6-2: Sorting option for languages, resolutions and template sets

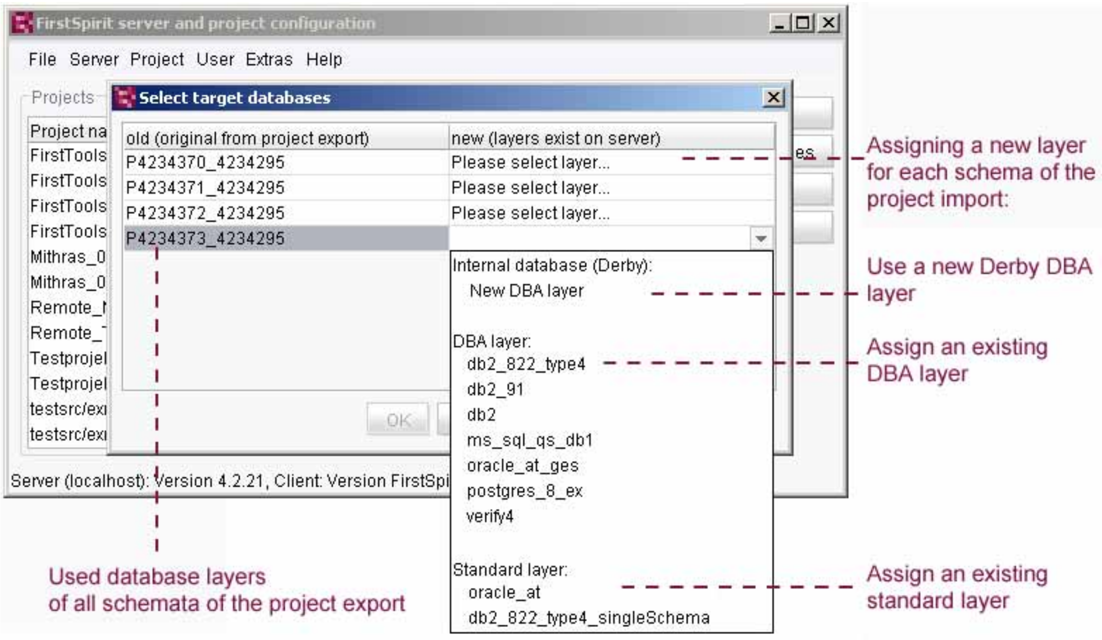
The changed order then affects the order of the tabs or the sorting of the resolutions in the project. The following applies to resolutions and languages: The ORIGINAL resolution and the master language always remain in first place in the order.

6.1.3 Enhanced import function for database layer

The export/import function for projects in the server and project configuration can now be used to assign a new or existing database layer on the server to each schema of a project.

Until now, when a project was imported it was only possible to assign one database layer for all the schemata of a project. Each schema of the export file are now displayed in the "Selection of the destination for databases" dialog (left-hand area). The new assignment of a layer for the imported project can be selected from the right-hand area. For each individual schema, either:

- a new Derby DBA layer,
- a default layer or
- a DBA layer can be selected.



old (original from project export)	new (layers exist on server)
P4234370_4234295	Please select layer...
P4234371_4234295	Please select layer...
P4234372_4234295	Please select layer...
P4234373_4234295	

Internal database (Derby):
New DBA layer

DBA layer:
db2_822_type4
db2_91
db2
ms_sql_qs_db1
oracle_at_ges
postgres_8_ex
verify4

Standard layer:
oracle_at
db2_822_type4_singleSchema

Server (localhost): Version 4.2.21, Client: Version FirstSpi

Assigning a new layer for each schema of the project import:

Use a new Derby DBA layer

Assign an existing DBA layer

Assign an existing standard layer

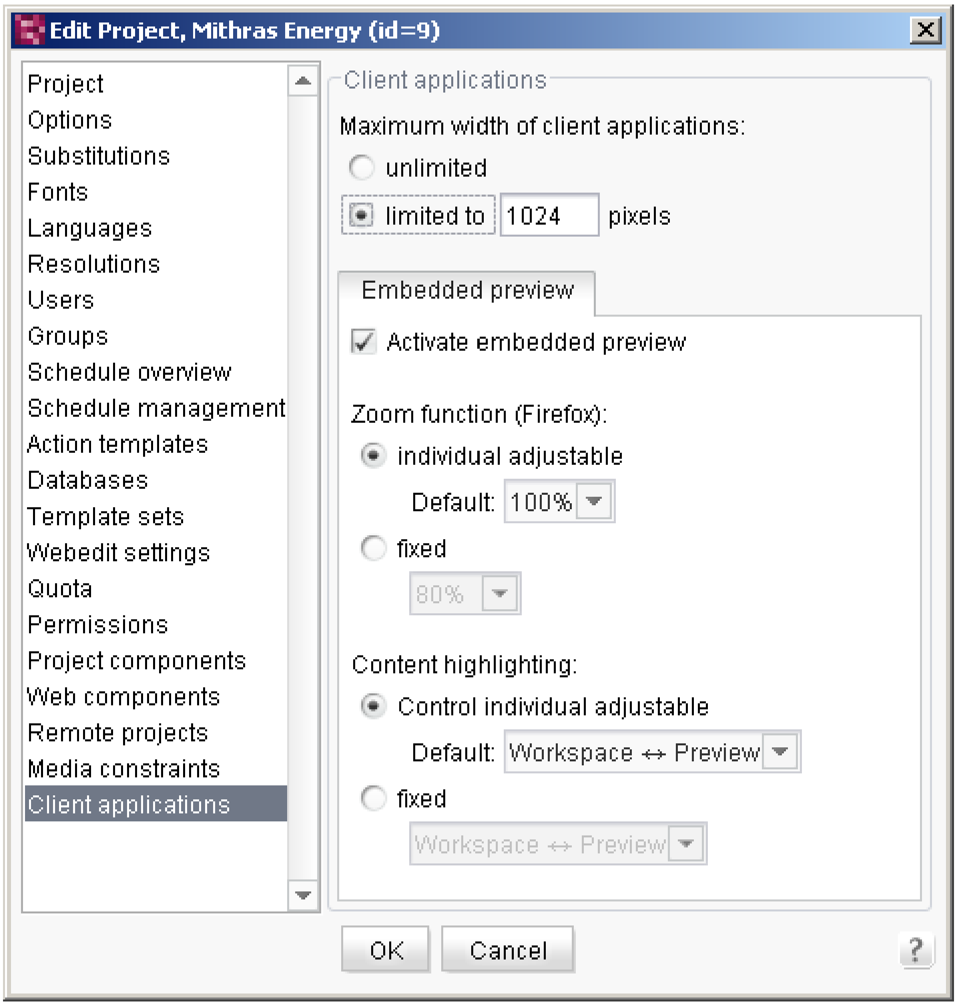
Used database layers of all schemata of the project export

Figure 6-3: Conversion of the database layer on importing a project

The choice of layer types decides whether FirstSpirit users can independently create new schemata in the project following the import (possible for DBA layers) or not (is prevented for default layers) (for a further description, see Chapter 8.1.6 page 116).

6.1.4 Project-wide pre-configuration of the Integrated preview

For each project the behaviour of the Integrated preview (siehe Kapitel 4.1.4 Seite 25) can be pro-configured in the project properties (Server and Project Configuration):



Edit Project, Mithras Energy (id=9)

Client applications

Maximum width of client applications:

○ unlimited

● limited to 1024 pixels

Embedded preview

☑ Activate embedded preview

Zoom function (Firefox):

● individual adjustable

Default: 100%

○ fixed

80%

Content highlighting:

● Control individual adjustable

Default: Workspace ↔ Preview

○ fixed

Workspace ↔ Preview

Figure 6-4: Project-wide configuration of the Integrated Preview

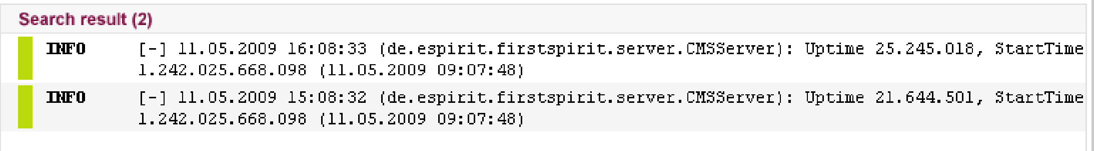
The settings made here have, among other things, an effect on the options which can be selected in the menu “View“ in the JavaClient.

First you can determine the maximum width of the Integrated preview using the radio buttons below **Maximum width of client applications**.

Via the option **Activate embedded preview** the use of the Integrated preview can be activated or deactivated for the entire project. If the option is activated, the scaling factor of the Integrated preview can be pre-configured via **Zoom function (Firefox)**. Via **Content highlighting** the control of the Content highlighting can be pre-configured. If the option **fixed** is selected the editors can no more change zoom factor and behaviour of the Content highlighting in the respective project.

6.1.5 Cyclical output of the VM start time and VM uptime

The log outputs of the FirstSpirit server were enhanced with FirstSpirit Version 4.2. The FirstSpirit server logs the VM StartTime and the VM Uptime as a one-off on starting and cyclically each hour. These outputs allow for example an improved interpretation of the GC logs (Garbage Collection), which contain only an offset since the start of the VM:


Search result (2)

```
INFO     [-] 11.05.2009 16:08:33 (de.espirit.firstspirit.server.CMSServer): Uptime 25.245.018, StartTime
         1.242.025.668.098 (11.05.2009 09:07:48)
INFO     [-] 11.05.2009 15:08:32 (de.espirit.firstspirit.server.CMSServer): Uptime 21.644.501, StartTime
         1.242.025.668.098 (11.05.2009 09:07:48)
```

Figure 6-5: FirstSpirit server monitoring: Log VM start time and VM uptime

6.2 Enhancements in FirstSpirit modules

6.2.1 "FirstSpirit EnterpriseBackup" module (data backup)

With its project export/import function, FirstSpirit already has a reliable, well tried and tested process with which projects can be backed up and restored, completely, consistently and while the system is running. A complete data backup of the whole project at short intervals, however, uses up large amounts of resources and in many cases is unnecessary as, for example, not all the revision-based data from the repository has to be regularly backed up.

New data backup strategies are therefore introduced in FirstSpirit Version 4.2, which are specially tailored to the requirements of extensive enterprise projects:

- Shortest possible runtime
- Avoiding restrictions to editing operation
- Integration in commercial backup solutions

To this end, a new "Enterprise Backup" module is made available with FirstSpirit Version 4.2. The module enables efficient data backup using incremental and differential backups. To use as little storage space as possible for the data backup, the new backup strategies no longer include a complete backup. From a defined starting point ("snapshot"), all changes to the project are now only backed up separately. If necessary, a complete backup can be created from the initial backup file ("snapshot" export) and the respective change backup files.

The different backup strategies can to a large extent be automatically performed by the system and suitably combined with each other. Performance of the data backup is configured (with the help of a Wizard) using the schedule management and then performs the required backups at the defined times, for example, hourly incremental and daily differential backups.

The module includes the following functions:

- **"SnapShot" backup:** The task of the "snapshot" backup is to backup the data of the whole project WITHOUT the historic data (from the revision-based repository). Only the current status and the last released status of all objects is backed up. A "SnapShot" backup therefore restores the complete project, but NOT the history of the project's elements and therefore the possibility of continuous tracking of changes. This data is only backed up by incremental or differential backups. A "SnapShot" export always forms the reference point for the restoration of all other project data (via differential or incremental backups).
- **Differential backup:** In a differential backup, all project data which lies between the revision of the last "SnapShot" export and the current revision of the project is backed up. Unlike a "SnapShot" backup, a differential backup also contains all historic states and deleted objects (which lie between the given revisions).
- **Incremental backup:** Analogous to the differential backup, an incremental backup saves the current project data and the historic data. The starting point for the data backup is always the revision of the last export. Unlike the differential backup, it does not necessarily have to involve a "SnapShot" export. The decisive difference between an incremental and a differential backup is that incremental backups build on each other. Therefore, in order to restore all project data, all the required export files ("SnapShot" backups, if applicable differential backups and all subsequent increments) must be available.
- **Backing up a FirstSpirit server:** The backup of a FirstSpirit server involves backing up all the server data, in particular the configuration settings. The project data and archive data are not included in the server backup.

Depending on the scope of the required backup, these strategies can be very time-consuming and resource intensive. In order to avoid limiting editing operations during the data backup, the schedules can be swapped out to one or several **separate FirstSpirit backup server**s (members of a cluster).

Apart from the pure data backup, **other use scenarios** are also possible for the "FirstSpirit Enterprise Backup" module:

- As an alternative to the archive function, the "FirstSpirit Enterprise Backup" can be used to **reduce the project size**. In this case, the data backup strategies are used so that a specific period of the project is completely exported (by combining a "SnapShot" export with a differential export and a final incremental export). The project can then be restored based on these export files. It now only consists of the current project data and the historic data of the export files.

! *Further information about the use of the module "FirstSpirit Enterprise Backup" see also the following Chapter 6.2.2.*

6.2.2 Long-term archiving and backup in FirstSpirit

The architecture of the data repository of FirstSpirit is designed for a complete traceability of all changes made. For this reason no data is deleted on principle, but they are only marked as "deleted" (analogue: changing or editing data corresponds to creating a new revision of the data). Especially in projects with many data and many changes this will lead to very large mounds of data. For this reason, FirstSpirit offers, besides an efficient architecture with a high stability for bulk data mechanisms for controlling these mounds of data: archiving (default; see *FirstSpirit Manual for Administrators,* Chapter "Project archiving") and Enterprise-Backup (module, see Chapter 6.2.1).

By means of the **Archiving** data which are no more required are moved from a FirstSpirit project into an archive repository (examples: deleted media or older versions). Attention should be paid to the fact that the archiving can potentially take a long time, and this process requires additional der system resources too. For this reason, the archiving process should take place continuously and should not be started shortly before achieving the limit of the hard disk. The archived data are no more available directyl for the editorial work (e.g. „restore older version"), but they must be removed from the archive into project by the project administrator, if required. The format of the archive repository is selected in such a manner that it need not to be changed (e.g. in case of FirstSpirit updates). Attention must be paid here to the fact that the content of the archive will not be converted in any way! Thus, it can principally not be guaranteed that data which have been restored from the archive can be used and edited correctly in the current project context / software version. This applies in particular for FirstSpirit structures with dependencies or relations to other (possibly external) systems and / or modules (e.g. LDAP server, database etc.) too.

❗ *ATTENTION: Project archiving is used for reducing the required space of a FirstSpirit project and DOES NOT realise an auditing-proof archiving in a legally binding sense!*

By means of the **Enterprise-Backup** module the operational reliability of FirstSpirit can be increased also in environments with large data volume and frequent changes. Different methods are available to this end, from the complete backup of FirstSpirit (including all modules and the Server and Project Configuration) to SnapShots to diffential and incremental methods. When operating the backup infrastructure you must always keep in mind that generally there are dependencies to external systems, which must be taken into account by the data backup and/or disaster recovery concept, if required. Especially in the case of FirstSpirit projects which are subject to structural changes (e.g. in the area of database modelling) a backup of the project on the basis of SnapShots is very important to avoid problems with incompatible database schemes.

❗ *ATTENTION: The Enterprise-Backup module (also on server level) is used for increasing the operational reliability of a FirstSpirit system and DOES NOT realise (without additional measures) an auditing-proof archiving in a legally binding!*

7 Installing FirstSpirit Version 4.2

7.1 Installation under Windows/Unix

Due to the use of Java, FirstSpirit is to a large extent a platform-independent client-server system. The installation essentially concerns the FirstSpirit server only, as the client is either managed via Java Webstart (JavaClient) or is operated as a web application via a web browser (WebClient). Due to the platform independence, an installed FirstSpirit server has a uniform layout in the file system structure on all operating systems. Differences only occur during the course of the installation. All the FirstSpirit server files, apart from several files necessary for the system start, are installed in the target directory.

The FirstSpirit server in Version 4.2 is available for the following operating systems:

- Red Hat Enterprise Linux
- Debian/GNU Linux
- SUSE Linux Enterprise Server
- LSB-conforming Linux distributions
- Solaris
- AIX
- Windows

All operating systems are supported in 32 and 64 bit architecture.

For further information about installation, please refer to "FirstSpirit Installation Instructions Version 4.2".

Each installation of FirstSpirit Version 4.x installs precisely one FirstSpirit server. In the event of renewed installation, the existing FirstSpirit server is updated – i.e. if a FirstSpirit installation, Version 4.1 already exists, when FirstSpirit Version 4.2 is installed the existing server is updated from Version 4.1 to Version 4.2.

By contrast, if it installed under Windows operating systems, an explicit differentiation is made between an update and a new installation:

- If FirstSpirit Version 4.2 is installed as a new installation or re-installation, a new FirstSpirit server is installed (with separate service, Start menu, etc.).
- If an update is installed, the existing installation is updated.

Note about uninstallation under: If a FirstSpirit server is to be uninstalled which has been installed with a FirstSpirit version between 4.2.1 and 4.2.19 (inclusively) under Windows by means of the file “firstspirit-setup.exe“ (see FirstSpirit Installation Instructions, Chapter 3.1), a FirstSpirit update must be executed before by means of the installer. For this purpose, the file “firstspirit-setup.exe“ must be used as well. The file should be of a version newer than 4.2.19. If no update is executed before or only by replacing the JAR file “fs-server.jar“ an uninstallation can potentially lead to deletion of entries of other software from the start menu, which had not been added during the FirstSpirit installation.

This applies not only to FirstSpirit servers which had been installed under Windows by means of the file “firstspirit-setup.exe“ (versions 4.2.1 to 4.2.19) but also server installations which had been updated to higher versions afterwards by replacing the JAR file.

7.2 For people changing over from FirstSpirit Version 4.1

The aim of the product development is make it as easy as possible for the users of FirstSpirit to changeover to the respective current version. In general it is possible to update a FirstSpirit server by exchanging a single file only ("In place Update" to a new build within a minor version line).

For more information about updating FirstSpirit servers see “FirstSpirit Installation Instructions Version 4.2“.

7.2.1 Update from FirstSpirit Version 4.1 to Version 4.2

FirstSpirit Version 4.2 is a "minor version upgrade" which can be implemented without having to make changes to the projects.

Several new functions are introduced with FirstSpirit Version 4.2, which require migration of the existing data (see Chapter 8.1 page 113).

Note: Adjustments may be necessary in individual cases where unsecured access to internal FirstSpirit structures takes place without using the FirstSpirit Access API!

Note: An update from FirstSpirit Version 4.1 to Version 4.2 is possible without making changes to the projects. By contrast it is not possible to downgrade from Version 4.2 to Version 4.1 (see Chapter 7.2.2 page 111).

Note: If modules are used they should be updated to the newest 4.2 version.

7.2.2 Downgrade from FirstSpirit Version 4.2 to Version 4.1

A downgrade from FirstSpirit Version 4.2 back to Version 4.1 is only possible with manual adjustments and is not recommended. Problems occur with the Lucene search index, the generic link editors, the content areas and on accessing remote projects:

- **Search index:** Due to the Lucene update from Version 2.1.0 to Version 2.3.2, the search index of new projects created in FirstSpirit Version 4.2 or already migrated into 4.2, is not downwards compatible. When the project is opened in FirstSpirit Version 4.1, an error message appears: "couldn't create index for project". In this case the index file in the `data/projects/project_${ID}/index` directory on the FirstSpirit server must be deleted. The project must then be re-indexed using a script. This is not possible while the server is running.
- **Generic link editors:** The introduction of "generic link editors" in FirstSpirit Version 4.2 also helps to migrate existing projects into FirstSpirit Version 5.0. Migration of the link templates and migration of the data based on these templates is supported by FirstSpirit. (see Chapter 8.1.2 page 114). This change is not downwards compatible. These changes must be manually reset on downgrading from FirstSpirit Version 4.2 back to Version 4.1.
- **Content areas:** The definition of content areas will be changed over with FirstSpirit Version 4.2 (cf. Chapter 5.1.12 page 96). Adjustment of the existing templates to the new syntax takes place automatically when the template is saved in FirstSpirit Version 4.2 (see Chapter 8.1.5 page 115). This change is not downwards compatible. These changes must be manually reset on downgrading from FirstSpirit Version 4.2 back to Version 4.1.
- **Remote access via FirstSpirit input components:** The configuration of remote projects within the FirstSpirit input components: CMS_INPUT_FILE, CMS_INPUT_PICTURE and CMS_INPUT_PAGEREF has been changed in 4.2 (see Chapter 5.2.2 page 99). Adjustment of the existing templates to the new syntax takes place automatically when the template is saved in FirstSpirit Version 4.2 (see Chapter 8.1.3 page 114). This change is not downwards compatible. These changes must be manually reset on downgrading from FirstSpirit Version 4.2 back to Version 4.1.

❗ ***Exporting** projects which have been created with a FirstSpirit server of a higher version, e.g. 4.2, and **importing** them subsequently on a FirstSpirit server of a lower version, e.g. 4.1, is not a guaranteed product feature! Technically, such an export/import is possible but can, especially in the case of an export from 4.2 to 4.1, lead to unforeseeable problems because of the many newly implemented input components, methods and incompatible changes in the software behaviour.*

8 Release Management

8.1 Announcements for FirstSpirit Version 4.2

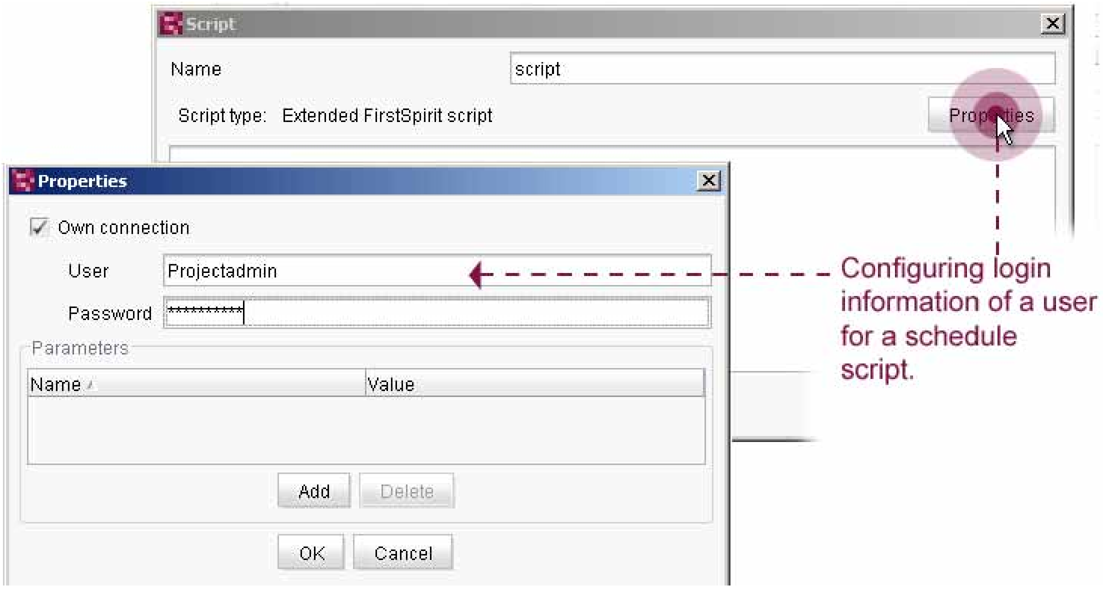
8.1.1 Changes to the FirstSpirit security model

Changes were made to the security model for FirstSpirit Version 4.2.

All changes to the project properties can only be made by server or project administrators.

The consequence of this change is that schedule scripts which make changes within the project or server properties also have to work on a corresponding user-specific connection.

Background: The schedule management of FirstSpirit can be used to deposit scripts which perform certain actions within a project or directly on the server. Properties can be configured for these schedule scripts, for example, the login information of a user, in order to set up their own connection with the server for executing scripts. Scripts which are configured without special user information continue to have read access, but cannot perform any modifying operations (cf. FirstSpirit Release Notes 4.1).



Name script

Script type: Extended FirstSpirit script

☑ Own connection

User Projectadmin

Password ***********

Configuring login information of a user for a schedule script.

Figure 8-1: Configuring a user-specific connection for a schedule script

Change in FirstSpirit Version 4.2: If actions modifying the project properties are performed in a script (within a schedule), this script must either be executed in the user context of a server administrator or of a project administrator.

8.1.2 Migration for link editors

From FirstSpirit Version 4.2, the configuration options for links are considerably enhanced by the introduction of generic link editors. The configuration of link templates can then, analogous to page and section templates, be created by adding input components into the form area (see Chapter 5.1.2 page 77).

Whereas static and generic link editors can still be used in parallel in FirstSpirit Version 4.2, **static link editors will no longer be supported from FirstSpirit Version 5.0**. The introduction of "generic link editors" in FirstSpirit Version 4.2 therefore also helps to migrate existing projects to FirstSpirit Version 5.0.

Software supported conversion of the templates: Conversion of the existing link templates to the new, generic link editors is supported by FirstSpirit. The automatic conversion can be started on the existing link templates using the context menu (Extras – Convert Link Template). With this, the static input fields used to date are replaced by the new input components in the form area. Project-specific manual adjustment of the layout may be necessary following the automatic conversion.

Software supported conversion of the existing data: The links maintained in the project to date, which were based on the static editors, are also automatically converted when saved again. In this way, all existing links should be able to be converted during operation of FirstSpirit Version 4.2. Additional manual work should not be necessary for converting the existing data.

Note: The migration of the link editors must be fully completed in FirstSpirit Version 4.2. Otherwise the projects cannot be migrated to FirstSpirit Version 5.0.

Note: These changes are not downwards compatible (see Chapter 7.2 page 110).

Note: The changeover to generic link editors changes the internal data structure of DOM elements. This can potentially lead to problems with existing scripts. The scripts must be manually adjusted if necessary.

8.1.3 New data element CMS_INCLUDE_OPTIONS

In FirstSpirit version 4.2 the new data element CMS_INCLUDE_OPTIONS is introduced. This data element can be used for configuring the set of values which are provided by an input component offering several values for selection dynamically (see Chapter 5.1.9 page 88).

CMS_INCLUDE_OPTIONS expands the options of the data element *CMS_INCLUDE_CONTENT*. In FirstSpirit version 4.2 *CMS_INCLUDE_CONTENT* can be used parallel with the new data element *CMS_INCLUDE_OPTIONS*. From FirstSpirit version 5.0 the data element *CMS_INCLUDE_CONTENT* will no more be supported.

8.1.4 Enhancement for input components with remote media access

The licence-dependent FirstSpirit remote media function has been enhanced for the input components CMS_INPUT_FILE, CMS_INPUT_PICTURE and CMS_INPUT_PAGEREF. Unlike FirstSpirit Version 4.1, in Version 4.2 it is now possible to define several remote projects (previously precisely one) for an input component. To this end, new tags for configuring the input components within the form area were introduced (see Chapter 5.2.2 page 99).

Automatic adjustment of the templates: Existing templates are automatically adjusted to the new syntax by FirstSpirit. On saving a template (from Version 4.1) in FirstSpirit Version 4.2, the configurations concerned are automatically converted to the new syntax. Manual adjustments are not usually necessary.

Note: These changes are not downwards compatible (see Chapter 7.2 page 110).

8.1.5 Definition of content areas via the template properties

From FirstSpirit Version 4.2, input options for defining content areas and section restrictions are available on the "Properties" tab of a page template (see Chapter 5.1.12 page 96). The definition of content areas used to date in the header area of a page template and the definition of section restrictions by using Drag & Drop to drop section templates onto a content area is no longer supported from FirstSpirit Version 4.2.

Automatic adjustment of the templates: FirstSpirit automatically adjusts existing content areas in the header area of a page template and the defined section restrictions of a page template to the new definition in the "Properties" tab. On saving a template (from Version 4.1) in FirstSpirit Version 4.2, the content areas and section restrictions concerned are automatically converted into the new definition. Manual adjustments are not usually necessary.

Note: These changes are not downwards compatible (see Chapter 7.2 page 110).

8.1.6 Renaming the layer types for database link

In FirstSpirit, the term "Layer" is used to describe the connection configuration to a Database Management System (DBMS). In FirstSpirit, a layer can either be assigned to precisely one or to several FirstSpirit schem*ata*. Accordingly, two different "layer types" are supported by FirstSpirit.

The name of the layer types for the database link in FirstSpirit 4.0 led to misunderstandings during use in the past. Therefore, the name is changed with the introduction of FirstSpirit Version 4.2. The familiar function is retained – it is not necessary to adjust existing projects.

Version 4.0 and 4.1	from Version 4.2	Description
Multi-project layer	**Default layer**	This layer type contains an explicit definition of the DB schema used in the layer definition. In this case, all tables of the FirstSpirit schemata which use this layer are stored in the given DB schema. In a FirstSpirit project to which default layers only are assigned, a FirstSpirit user cannot create any new additional schemata. Only the FirstSpirit administrator can add further default layers to the project. A default layer should always be assigned to precisely one FirstSpirit schema only.
Single project layer	**DBA layer**	This layer type is a new introduction in FirstSpirit Version 4.0 and does not contain any explicit definition of the DB schema to be used. FirstSpirit automatically creates a separate DB schema for each FirstSpirit schema. With this, FirstSpirit users are now also able to create further schemata. However, for most DBMS, this requires comprehensive DBA rights (DBA = Database Administrator).

8.1.7 New menus in FirstSpirit JavaClient

The new "View" menu in JavaClient was also introduced with the new functions for FirstSpirit Version 4.2.

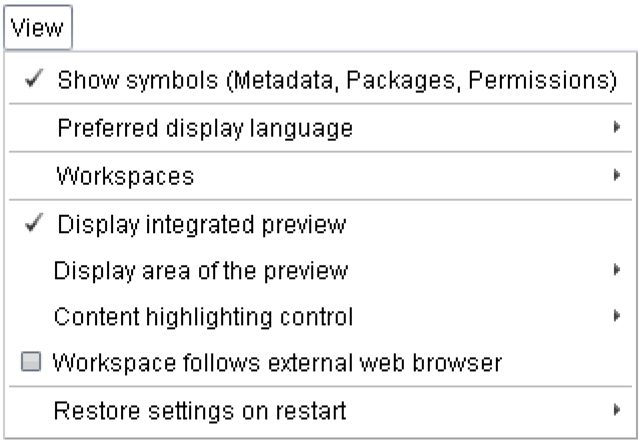

View
✓ Show symbols (Metadata, Packages, Permissions)
Preferred display language ▸
Workspaces ▸
✓ Display integrated preview
Display area of the preview ▸
Content highlighting control ▸
□ Workspace follows external web browser
Restore settings on restart ▸

Figure 8-2: View menu

The menu provides configuration options for the new FirstSpirit functions:

- Workspaces (see Figure 4-4)
- integrated preview (see Figure 4-14)
- Content Highlighting (see Figure 4-22)
- Restore settings on restart (see Figure 4-42)

In addition, several menu entries from the "Extras" menu have been adopted in the new "View" menu:

- Show symbols
- Preferred display language

8.1.8 Enhancement of Access API: DomElement

For the return type `DomEditorValue`, a new API interface `DomElement` has been introduced. Scripts which use the previously unstable "`DomElement` "class" must be adjusted to the new interface:

Example - Generating content in the DOM Editor (to date – unstable):

```
final DataValue dataValue = data.get("cs_payment_text");
final DomEditorValue editor = (DomEditorValue) dataValue.getEditor();
xmlBuf = new StringBuilder();
xmlBuf.append("<p>myDom</p>");
editor.set(language, DomElement.fromXml(xmlBuf.toString()));
```

Example - Generating content in the DOM Editor (from FirstSpirit Version 4.2):

```
final DataValue dataValue = data.get("cs_payment_text");
final DomEditorValue editor = (DomEditorValue) dataValue.getEditor();
xmlBuf = new StringBuilder();
xmlBuf.append("<p>myDom</p>");
final DomElement domElement = editor.get(language);
domElement.set(xmlBuf.toString());
editor.set(language, domElement);
```

Note: The changes are downwards compatible. After adjusting in Version 4.2, the scripts also function in FirstSpirit Version 4.0 and 4.1.

8.1.9 Changes to the function "Change Project" (Menu: Project)

The menu item "Change Project" in the JavaClient is deactivated at the initial release date of FirstSpirit Version 4.2, because it cannot be assured that the function works properly. If this function will be made available in a later build again or if it will be dropped permanently can not be determined conclusively at the moment. For further information about the status of this menu item please refer to subsequent FirstSpirit Release Notes.

8.1.10 Changes to the function "Reconnect"

In case of an interruption of the communication between FirstSpirit JavaClient and FirstSpirit Server the function „Reconnect" has could been used for re-establishing the connection to the server without stopping the JavaClient. This menu item is not available at the initial release date of FirstSpirit Version 4.2, because it cannot be assured that the function works properly. If this function will be made available in a later build again or if it will be dropped permanently can not be determined conclusively at the moment. For further information about the status of this menu item please refer to subsequent FirstSpirit Release Notes.

8.1.11 Changes to the default behaviour of newly created projects

The options „Permission check“ and „Use Release function“ had been inactive by default when creating a new project and had to be activated manually if required. From FirstSpirit version 4.2 these options are active by default. Existing scripts, which allow to create automatically new projects, must be adjusted appropriately.

8.1.12 Notes about the use of virus scanners

When operating the FirstSpirit JavaClient (in the HTTP mode) in system environments with activated internet virus scanner delays in the communication between JavaClient and FirstSpirit Server could be observed (arisen with NOD 64 Bit V4.0 under Windows 7). In the socket mode, no delays have been observed, so far. In case of problems you should optionally

- start the JavaClient in the socket mode
- change the configuration of the virus scanner
- use a new virus scanner.

8.2 Expiring functions in FirstSpirit Version 4.2

The following functions have been removed within the scope of the release plan for FirstSpirit Version 4.2:

- **Classic look & feel in JavaClient:** FirstSpirit JavaClient and the application for server and project configuration could be optionally run in FirstSpirit Version 4.1 with either the new Look & Feel or the previous, classic Look & Feel. The classic "look & feel" is no longer available from FirstSpirit Version 4.2.
- **Theming will be limited in WebClient:** The theming function is now only usable to a limited extent in WebEdit 4.2 (and will be completely dropped with WebEdit 5.0). Among other things, the toolbar in the XP Theme has been completely revised (see Chapter 4.2.1 page 67). The toolbar is therefore no longer included in the XP Theme. The following applies to all other themes: All css files from the theme project have been manually revised and are no longer contained in the theme project. The same applies to new (and changed) input components in WebEdit 4.2 (among other things, CMS_INPUT_OBJECTCHOOSER, FS_REFERENCE). These components are no longer contained in the Theme project.
- **Access 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) will be removed.
- The **compatibility mode for FirstSpirit version 3.1** is dropped in FirstSpirit version 4.2. For this reason, the option "Copy all media in this folder during generation" within the Media Store can no more be used. This function is replaced by the option to swap out the generation of media into a separate schedule (see FirstSpirit Release Notes version 4.1). The related API methods should no more be used in FirstSpirit version 4.2 (status "deprecated") and will be dropped in FirstSpirit version 5.0. When editing projects of the FirstSpirit version 4.1 an appropriate warning (*WARN*) will be output in the log file.
- The option **MultiViews path** in the project properties is dropped in FirstSpirit version 4.2. The related API methods are set to the status "deprecated". The method for the creation of paths and the creation of the MultiView paths can be selected in the action “Execute generation” (option “PathGeneration“) in generation tasks (see *FirstSpirit Manual for Administrators*, Chapter 7.5.9.2 “Execute generation“). When editing the project appropriate warnings (*WARN*) will be output in the log file.

8.3 Announcements for FirstSpirit Version 5.0

WebEdit 5.0 will be available with FirstSpirit V 5.0. WebEdit 5.0 does not involve simple revision or enhancement of the existing WebEdit function, but instead is a complete re-implementation with the help of the GWT[8] framework.

The usability improvements implemented with WebEdit 4.2 will be consistently and logically extended and further developed with WebEdit 5.0. WebEdit 5.0 effects further improvements to user prompting and usability and highly enhanced functionality.

The WebEdit Version 4.x used to date will be replaced by the new Version 5.0. This will require migration of existing projects.

WebEdit 5.0 will not have a theming function.

In general, no active components or plug-ins are necessary for use of the basic functions of WebEdit 5.0. But additional components have to be installed if advanced functions such as "Drag & Drop" are to be used. For example, simple uploading of a file within WebClient is possible without additional installations in the web browser. If, on the other hand, functions such as multiple selection or "Drag & Drop" are to be used in uploading, the corresponding applet may have to be installed.

Additional functions in WebEdit 5.0 will, if necessary, be made available as **Java Applets**.

[8] GWT – Google Web Toolkit

8.4 Expiring functions in FirstSpirit Version 5.0

The following functions have been removed within the scope of the release plan for FirstSpirit Version 5.0:

- **Theming in WebEdit:** The theming function will no longer be supported in WebEdit 5.0.

- **Static link editors:** Whereas static and generic link editors can still be used in parallel in FirstSpirit Version 4.2, the static link editors will no longer be supported from FirstSpirit Version 5.0. The introduction of "generic link editors" in FirstSpirit Version 4.2 therefore also helps to migrate existing projects to FirstSpirit Version 5.0 (see Chapter 8.1.2 page 114).

- **Data element CMS_INCLUDE_CONTENT:** The data element CMS_INCLUDE_CONTENT will no more be supported in FirstSpirit version 5.0. CMS_INCLUDE_CONTENT will be replaced by the enhanced data element CMS_INCLUDE_OPTIONS (see Chapter 5.1.9 Seite 88).

- **Access API – deprecations:** All functions of the FirstSpirit Access API, the discontinuation of which was announced in FirstSpirit Version 5.0 or before then ("deprecations", cf. API documentation) will be removed.