FirstSpirit™
Your Content Integration Platform

FirstSpirit Manual for Editors (JavaClient)
FirstSpirit Version 4.x

e-Spirit AG

Table of content

1 Introduction

1.1 Topic of this documentation

This documentation describes all relevant aspects for editors of the **FirstSpirit**™ **JavaClient** editing system. Functions required for creating and updating/maintaining editorial contents are explained. Administrative functions and aspects required for template development in FirstSpirit are explicitly excluded. These topics are explained in separate documents.

Based on the user interface of the FirstSpirit JavaClient and its screen view divided into two, in each chapter the elements of the tree structure (in the left-hand half of the screen) are described first and then the editing masks of the individual objects (right-hand half of the screen).
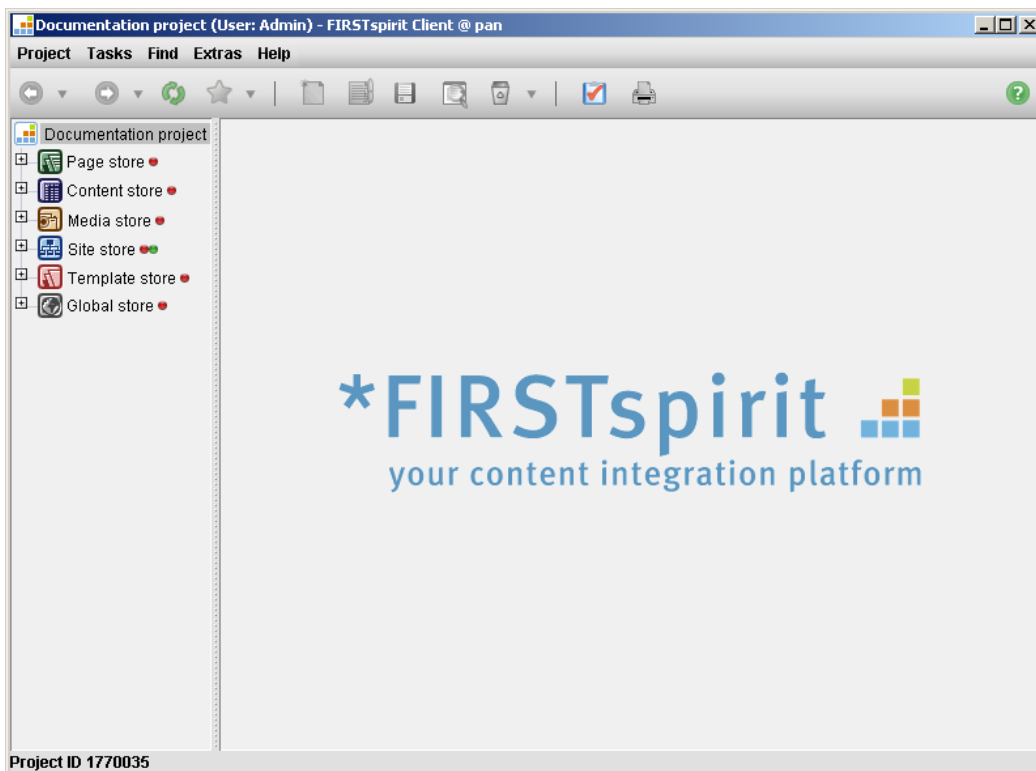
Figure 1-1: FirstSpirit JavaClient

> ❗ *From FirstSpirit Version 4.2 the user interface can be split into three parts. For details about the new functions and new layout see Chapter 1.3.2 page 25.*

The options available for starting JavaClient are explained in **Chapter 2**. (See Chapter 2 page 37)

In **Chapter 3** the functions of the menu bar and toolbar are described, in which general and project-wide functions are available. (See Chapter 3 page 48)

Chapter 4 contains descriptions of the Page Store for maintaining editorial content. All functions to be executed via context menus are explained in detail, as well as the input components available for editorial work (see Chapter 4, page 106).

Chapter 5 deals with the Content Store which is used for acquiring, entering and managing highly structured content. (See Chapter 5, page 133)

Chapter 6 describes the Media Store for managing all media used. Apart from all the functions to be executed via context menus the settings options at the individual levels are explained in detail. In addition, use of the Media Import Wizard is described. (See Chapter 6, page 160)

Chapter 7 deals with the navigation structure of a website displayed in the Site Store. Apart from all the functions to be executed via context menus the setting options at the level of a menu, page reference and document group are explained in detail. (See Chapter 7, page 215)

Chapter 8 is a general description of how the Template Store works. (See Chapter 8, page 244)

In **Chapter 9** the functions of the three areas: Global Content (maintenance of small components of pages), Project Settings (definition of replace rules) and User Settings (integrate Editors and Browsers) are described. (See Chapter 9, page 245)

Chapter 10 provides a detailed overview of the standard FirstSpirit input components (see Chapter 10, page 249).

Chapter 11 deals with the functions for general use of FirstSpirit (see Chapter 11 page 314).

The function of workflows is described in **Chapter 12** (see Chapter 12, page 422)

In **Chapter 13** the mechanisms provided by FirstSpirit for assigning and checking permissions are described and their specific use is outlined. A differentiation is made between permissions valid for a user of FirstSpirit, for example for an editor (editorial permissions), and permissions defined for a user of the site generated with FirstSpirit (user permissions) (see Chapter 12.4, page 433).

In practice you should familiarise yourself with general features first, such as the task list or user settings.

> *FirstSpirit is not a universal "out-of-the-box" product, but is software which is constantly being developed. New functions are continuously being integrated and customer ideas are realised. The documentation can only satisfy this constant updating process to a limited extent. It is therefore possible that a figure in this document differs from the current view in FirstSpirit due to slight time delays. Do not let it confuse you and follow the instructions as usual.*

1.2 The FirstSpirit concept

Extensive publications, for example the website of a company, involve a large amount of information which has to be managed, updated and published. FirstSpirit tries to make this as easy as possible by dividing the information into different "stores" (management areas) – while retaining the strict separation of layout, content and structure. Depending on the role assigned to them in the project, the user should keep an overview of the content and structure.

1.2.1 Stores

Within the FirstSpirit editorial environment different content (for example pictures, files, structured data, layout and navigation structure) are managed in separate areas (stores). This concept fulfils the paradigm of separation of structure, content and display of a website. The individual areas can be changed independently of each other and content can be reused at any time. This concept allows, for example, editors to make editorial changes easily, efficiently and above all without knowledge of HTML and XML.

FirstSpirit JavaClient is divided into six stores each with different colours, which are strictly separated from each other but are functionally dependent:
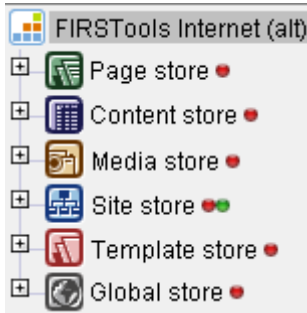
Figure 1-2: Stores in FirstSpirit JavaClient

The stores look like this in the new Look & Feel "LightGray" (from FirstSpirit Version 4.1):

Figure 1-3: Stores in the Look & Feel "LightGray"

View in FirstSpirit Version 4.2:
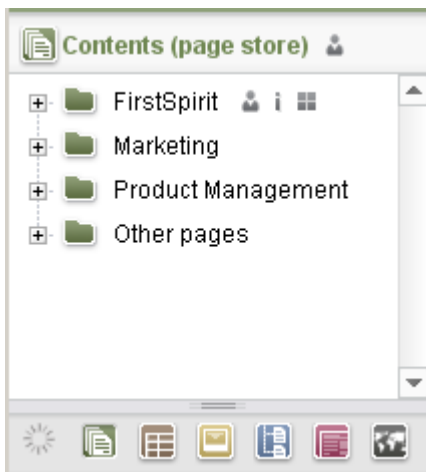
Figure 1-4: Stores in FirstSpirit Version 4.2

- The Page Store contains all pages and their editorial content (see Chapter 4, page 106).

- Content store is used to create highly structured pages by managing the content via database mechanisms
(see Chapter 4.4.10, page 128).

- The Media Store contains all media files used in the project. These do not necessarily have to be classic image files. The Media Store also manages other types of files, for example those that can be made available on a page for downloading (audio and video files, Flash animations, PDF documents, style sheets, etc.) (See Chapter 6, page 160).

- The Site Store determines the navigation structure of the website
(see Chapter 7, page 215).

- The Template Store includes all information concerning the layout of the website
(see Chapter 8, page 244).

- The Global Store contains global user and project settings and global content (with multiple use)
(see Chapter 9, page 245).

1.2.2 Permission assignment

In addition, FirstSpirit provides a well thought system of permission assignment, as a large website can only be effectively managed if each individual employee has precisely defined tasks. Each website created and maintained with FirstSpirit is called a project. A roles concept defines task-related access to parts of the system and describes the allocation of a team's work within the project. The permissions can therefore be intuitively assigned in easily understood and clearly followed roles. A permissions concept is created for each role and is then assigned to the relevant group of employees. When a user logs on in FirstSpirit with their name and password they are only given access to the system corresponding to the permissions assigned to them by virtue of their role. For example, an administrator is given all permissions for access to system settings, the chief editor access to the structure of the website and the editor is, for example, only given access to a special sub-area for maintaining the website (see Chapter 13.1.2, page 445).

FirstSpirit differentiates between permissions valid for a user of FirstSpirit, for example for an editor (editorial permissions), and permissions defined for a user of the page generated with FirstSpirit (user permissions).

- **Editorial permissions:** These are the permissions valid for a user of FirstSpirit (see Chapter 13.1 page 443).

- **Permissions to execute workflows:** These are a special type of editorial permissions which refer to the workflows within a project only (see Chapter 13.2 page 462).
- **User permissions:** Are permissions valid for the "user" of the site generated with FirstSpirit. User permissions are always linked with the personalisation system used (see Chapter 13.3, page 471).

1.2.3 Multilingualism

FirstSpirit consistently supports the concept of multilingualism which runs through all aspects of FirstSpirit:

Language dependent configuration options in the FirstSpirit JavaClient:
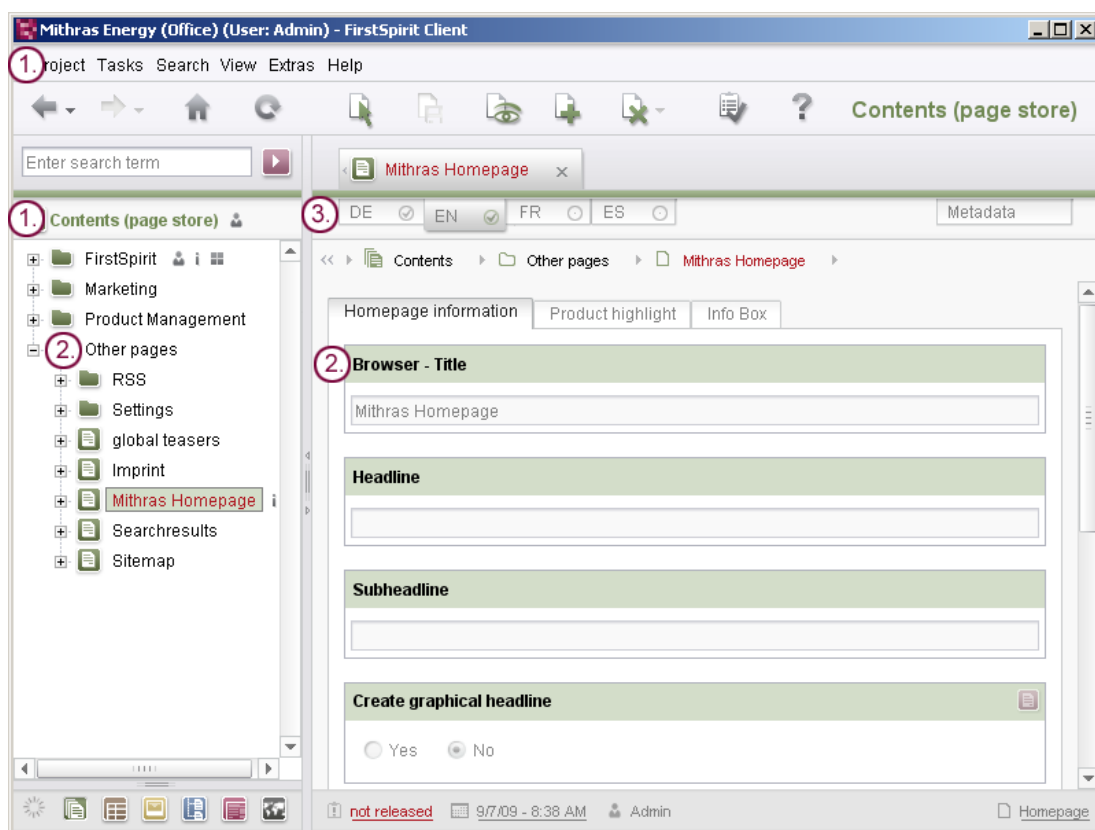
Figure 1-5: Language setting in JavaClient

1. **Language setting of FirstSpirit JavaClient ("Locale"):** This setting is defined using the combobox of the FirstSpirit Start page (see Chapter 2.2 page 39) and affects the language of both the Start page and the language of all FirstSpirit applications (including the JavaClient) started from the Start page. In JavaClient, the locale language determines the labelling of the menu

bar, the dialogs and all content which have not been deposited in the project in language-dependent form by the editor or template developer. Apart from German, English, French, Spanish and Russian, Italian can now also be selected from Version 4.1, Dutch from Version 4.2R2.

2. **Editing languages:** Editing languages are defined for a project by the project administrator and can then be configured by the editor using the "Extras – Preferred Display Language" menu, from FirstSpirit Version 4.2 the "View – Preferred Display Language" menu (see Chapter 3.1.4.2 page 69). The editing languages affect language-dependent contents which have been defined by the template developer, e.g. within the page or section templates. The relevant language-dependent labels are displayed to the editor in language-dependent form, for example, in the form area (labelling of the input fields, tooltips, elements of a combobox, etc.).

3. **Project languages (content languages):** The third setting option concerns the project languages, i.e. the languages defined by the project administrator for the inclusion of language-dependent editorial content. The editorial content can be entered in JavaClient using the language tabs of the Page, Media and Content Store.

   A generated site can be multilingual, i.e. the form area contains different input fields per project language, for example for texts and media, the contents of which are to be displayed depending on the chosen language ("language dependent").

   However, it is also possible to define "language independent" input fields. In this case the content is entered once only but is available in all project languages. This time-saving, language-independent definition of content is useful, for example, for the display of pictures (without text) or for numerical details (e.g. product descriptions, dimensions).

   Apart from simple editorial content, structured content from a database can also be integrated. The concept of multilingualism is taken into account in this case too.

In addition, FirstSpirit offers support for automated translation processes, for example:

- XML Export/Import (e.g. for Trados).

- Possibility of incremental translation using the "Page not yet completely translated" function.

- The option of a fallback, i.e. configurable, automatic language replacement. If the contents of a page are not available in a certain language the content can

be displayed in another language instead, for example, the project master language.

1.2.4   Parallel access in multi-user environments

FirstSpirit was conceived for use in multi-user environments. This means that, in particular, the editorial maintenance of the content can be carried out within a project in parallel by a large number of employees. Due to the close teamwork it is necessary to ensure that no conflicts occur during joint access to individual objects.

FirstSpirit differentiates between actions which may run on an object in parallel and actions during which access to an object has to be protected for the period of editing to prevent conflicts.

In FirstSpirit the following actions can be carried out in parallel, i.e. at the same time for an object:

- Create: Create new objects in the project.

- Delete: Delete already existing objects from the project.

- Copy: Copy already existing objects to another position within the project.

- Move: Move already existing objects to another position within the project.

The following actions may not be carried out in parallel:

- Change: Changing the contents of an object, for example a page of the Page Store.

- Release: Parallel access also has to be prevented when an object is released as the release ensures that only one precisely defined and checked status of an object is released.

The edit mode system realised in FirstSpirit is used to prevent the possibility of parallel editing. If a user chooses an object to be edited (e.g. a page of the Page Store on which content changes are to be made), they must block this object from access by other users for the duration of their work. Content changes to an object are only possible if the object is in edit mode. This ensures that several users do not work on an object simultaneously which would cause inconsistent data to be produced. This also achieves a high degree of security and simultaneously creates a possibility for working in parallel with a correspondingly high edit speed in multi-user environments.

1.2.5 Versioning, historizing and archiving

Versioning, historizing and archiving of all information has an important status in Enterprise Content Management. The primary objective is the most continuous traceability possible of all changes, as well as access to "system states from the past".

Versioning: Each time an object is changed by an editor, for example a medium, FirstSpirit creates a new version of this object. Thus, an object has a version history on the base of which it is possible to trace which changes were made by which persons over time. Apart from these simple changes, versioning in FirstSpirit also takes into account complex structures which can exist between the objects of a project. In FirstSpirit, e.g. pages are compiled from individual sections and are intertwined in the Site Store to form navigation. The versioning of the basic objects and the structural aspects gives a completely versioned description of the overall system status which enables changes to be traced.

Historizing: Historizing in FirstSpirit builds on this completely versioned description and is used to reinstate a system state from the past. The historizing can be used, for example, to generate the status of a website as of 01.01.2006 by (temporarily) setting the current live project to the status of 01.01.2006 and then carrying out a generation. Historizing generates a temporary state only and allows read access only.

Archiving: Archiving is used for long-term, secure and unadulterable storage of data. Within the scope of enterprise content management systems, archiving if frequently focussed on collating content in self-contained units and subsequent transfer to a long-term storage medium.

1.3 The FirstSpirit user interface

The FirstSpirit JavaClient is available with two display options: with the "Classic" Look & Feel and the new "LightGray" Look & Feel. The "LightGray" look & Feel is released from FirstSpirit Version 4.1 and is the default setting in this version. The classic "Look & Feel" is no longer available from FirstSpirit Version 4.2.

The view can be switched over in JavaClient using the "Extras" menu (Chapter 3.1.5.6 page 79) or using the connection settings on the FirstSpirit Start page (Chapter 2.2.4.1 page 28).

> ❗ *New functions released from FirstSpirit Version 4.1 are displayed in this document with the new Look & Feel; for the time being, existing functions continue to be displayed with the old Look & Feel.*

1.3.1   The FirstSpirit JavaClient with its new "Look & Feel" (from V4.1)

Each user now has a personalised project entry page (homepage) in FirstSpirit JavaClient. It can be opened by pressing the 🏠 button of the JavaClient toolbar or (up to Version 4.1) using the project start nodes in the tree view. It provides the user with a personalised overview of their bookmarks, open tasks and possible actions with corresponding icons and in the colour of the respective store.). Initially, only 10 tasks are displayed on the personalised project entry page. Click the "Display older tasks" button to show a further 10 tasks. Click "Display all tasks" to open the Task list.

Bookmarks are displayed with additional path details, open tasks with priority and the next required action (e.g. "check"). Open tasks are displayed in different colours: tasks which have been explicitly assigned to the editor are displayed with red lettering, tasks which the editor can open/insert, but for which they have not been explicitly entered as the modifier are displayed with black lettering.

With each click on the respective entry, further actions can be performed directly via the entry page. For example, one of the user's open tasks can be directly opened and forwarded using the link displayed after "Open tasks". (For details of open Tasks see chapter 11.1 page 314.)

A search window for full-text search within a project is now available directly in the main view of JavaClient (for details of this search dialog's function from FirstSpirit Version 4.2, see Chapter 3.3 page 106):

| Enter search term | ▶ |

The display of tabs has been changed: The user's current position within the hierarchy, e.g. in which folder, on which page, in which section, in which medium, etc. the user is currently located is now also clearly shown in the right-hand window. As part of this changeover, e.g. the "Page completely translated for this language" checkbox has been moved into the respective language tab. The checkbox can only be activated or deactivated in the active tab:

New icons have been designed for the FirstSpirit toolbar:

Adjustment of the icons and the colour guidance system for the individual store in FirstSpirit JavaClient (see Figure 1-3).

1.3.2 Revised user interface design from FirstSpirit Version 4.2

1.3.2.1 Editing in several workspaces ("multi-tabbing") (from V4.2)

Horizontal tab navigation within the FirstSpirit JavaClient editing area has been introduced with FirstSpirit Version 4.2.

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.
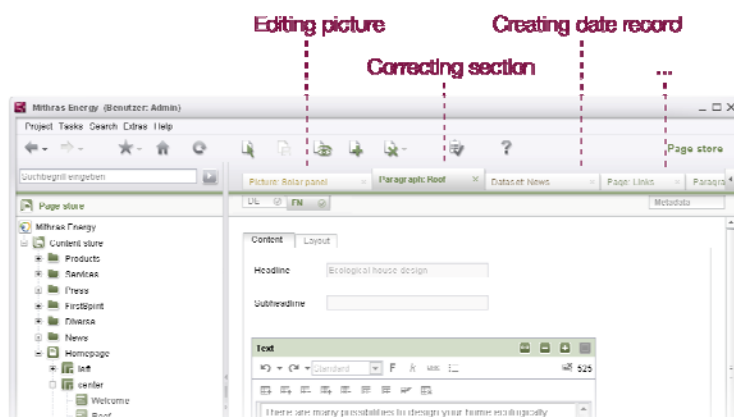
Figure 1-6: Multi-Tabbing

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:
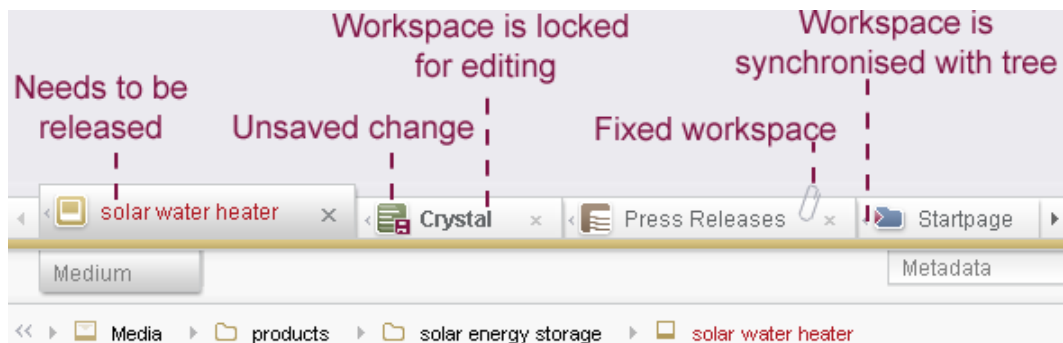
Figure 1-7: Display of fixed and unfixed workspaces

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.

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 1-7).

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 11.4 page 318).

The "**Restore Closed Workspaces**", which is opened via the context menu on each tab of an open workspace, can be used to re-open an already closed workspace (see also Chapter 3.1.4.3 page 70).
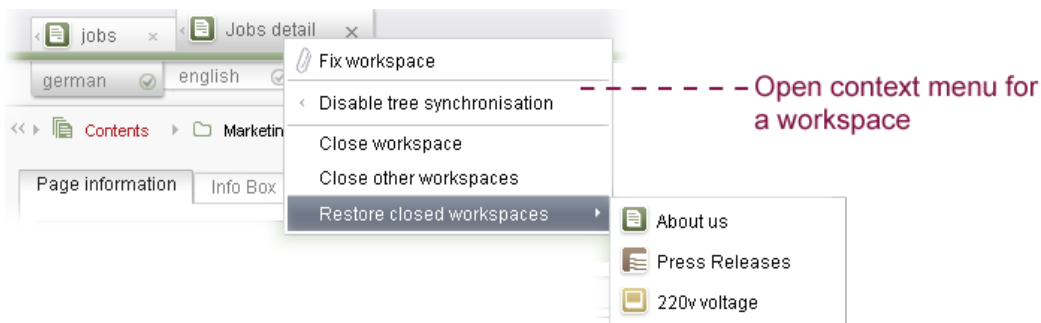
Figure 1-8: Context menu on a workspace

Working with workspaces can be configured for the editor – both globally for the whole JavaClient using the "View" menu (see Chapter 3.1.4.3 page 70) as well as locally for each individual workspace using the context menu of the respective tab (see Figure 1-8).

Automatic synchronisation of the active workspace in the right-hand area of the JavaClient with the tree display in the left-hand workspace (see also Chapter 3.1.4.3 page 70) can be activated or deactivated using the context menu of the respective tab (**"Activate/Deactivate Tree Synchronisation"** see Figure 1-8). 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:

▪ Navigation using the keyboard: If the user navigates using the keyboard (within the tree view), the current selection (visualised by a coloured background, e.g. power inverter) in the workspace and in the tree is initially retained; only the

focus within the tree view (visualised by a coloured frame, e.g.

⊞ 📁 power inverter) changes. The active workspace is therefore not affected by navigation through the tree elements using the keyboard, i.e. it does not change. 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 1-9).

  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 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.
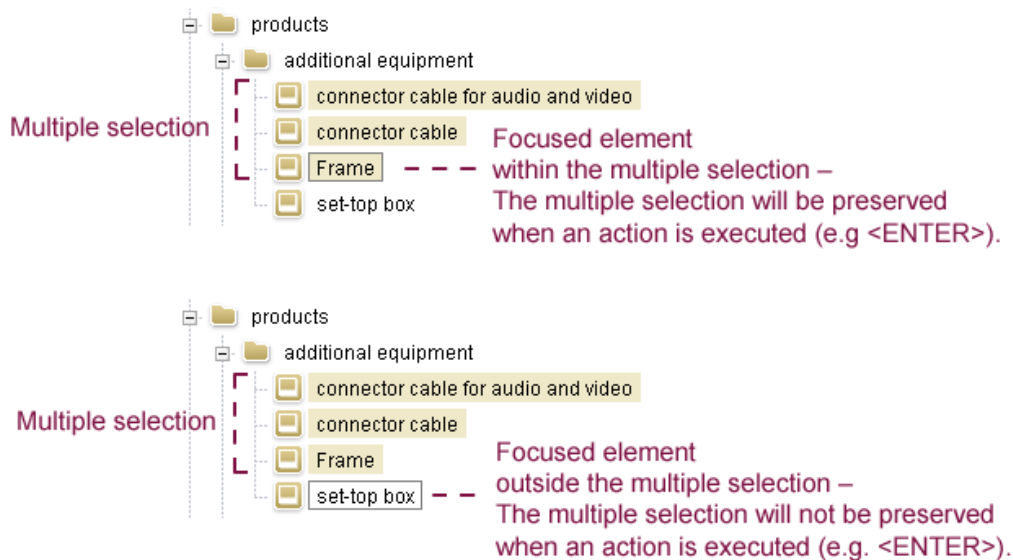
Figure 1-9: Multiple selection in the tree

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.

1.3.2.2 Breadcrumb Navigation (from V4.2)

From FirstSpirit Version 4.2, the path from the Store root up to the current element is displayed above the form area in the form of "breadcrumb navigation". This display is particularly helpful if synchronisation of the workspace with the tree is not enabled (see Chapter 3.1.4.3 page 70 and Chapter 1.3.2.1 page 25).

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 1-10).
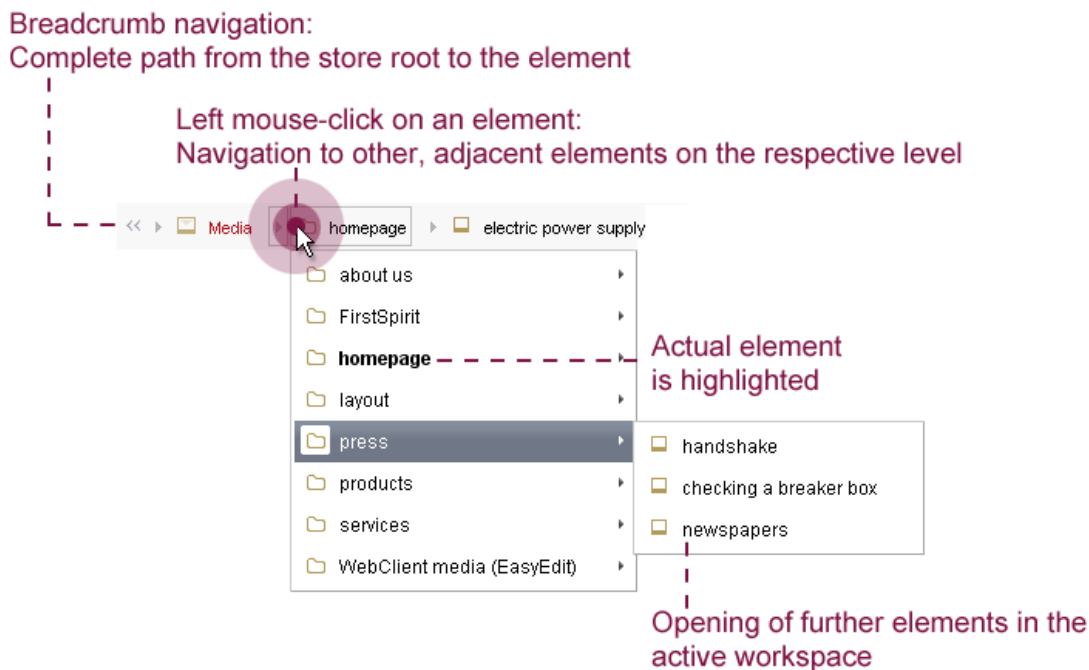
Figure 1-10: Navigation using the path elements ("breadcrumb navigation")

Analogous to opening the context menu in the tree, it can also be opened by right-clicking a path element, for example to start a release on the page which has just been edited.

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 1.3.2.1 page 25).

1.3.2.3 Display of the Stores as navigation areas (from V4.2)

As part of the introduction of "multi-tabbing", the overall project display in the left-hand navigation area of FirstSpirit JavaClient has been 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 1-4). When a button is clicked, the required store is shown as a navigation tree and the area open to date is hidden again.

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.

1.3.2.4 Integrated preview in JavaClient (from V4.2)

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:
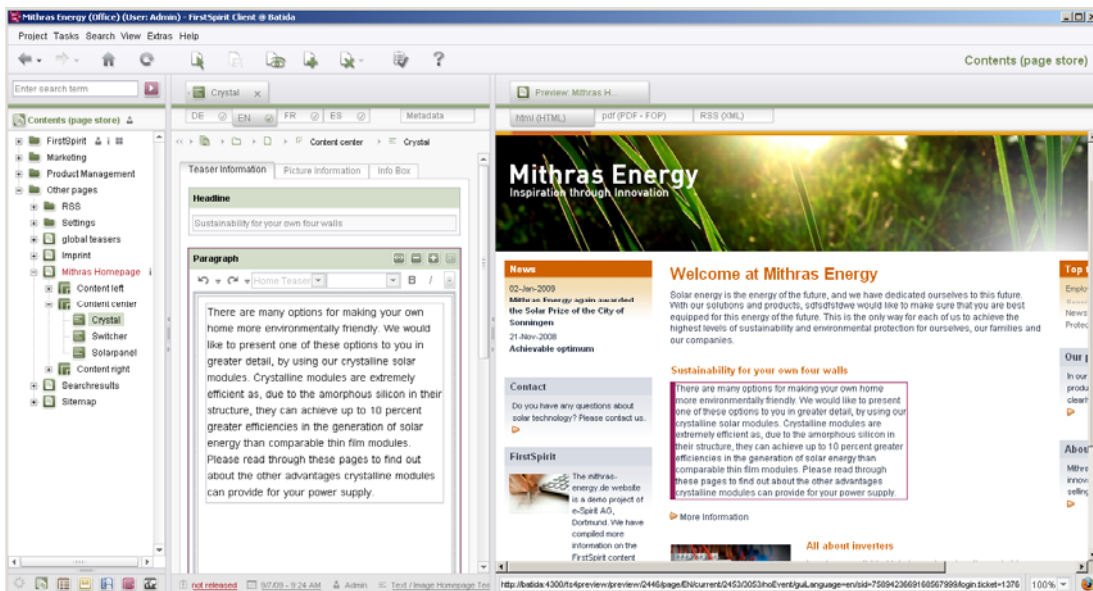
Figure 1-11: Display of the WYSIWYG preview

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. 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.

> **!** *The display of an output channel in the integrated preview may required the installation of further plug-ins. For example, the Acrobat Reader Plug-in must be installed on the editor's workstation and correctly configured in order to display an integrated preview in the PDF output channel.*

Representations of the datasets can be displayed within the page context in the Content Store. If a certain data set (data record) is selected in FirstSpirit's Content Store, use of the data set from the Page Store is displayed 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)

and depending on the area that is clicked in the integrated preview:

- Display component (click)
- Edit component (shift + click)
- Display section (click)
- Edit section (shift + click)
- Display content area
- Edit content area
- Display page (click)
- Edit page (shift + click)
- Display data set (click)
- Edit data set (shift + click)

The integrated preview is configured using the "View" menu (cf. Chapter 3.1.4.4 page 72).

In addition, a scaling factor of 50% to 200% can be defined in the bottom part of the preview (see Figure 1-12). 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.
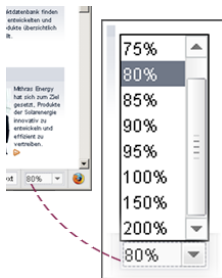
Figure 1-12: Setting the scaling factor for the integrated preview

> *Depending on the configuration by the project administrator in the project properties the selection can be inactive and a fixed scaling factor can be preset.*

> *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.*

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

> *Use of preview integration requires a 32 bit Java Runtime Environment (JRE) on the workstation computer. The browser integration is not compatible with a 64 bit JRE.*

1.3.2.5   Functional enhancements of the Interated preview

From 4.2R4, the "Integrated preview" introduced in JavaClient in the initial version of FirstSpirit 4.2 can not only be used for a preview of the currently edited content as to date, but also other file formats and the FirstSpirit Online Documentation can now also be displayed there, namely with the help of an application matching the format.
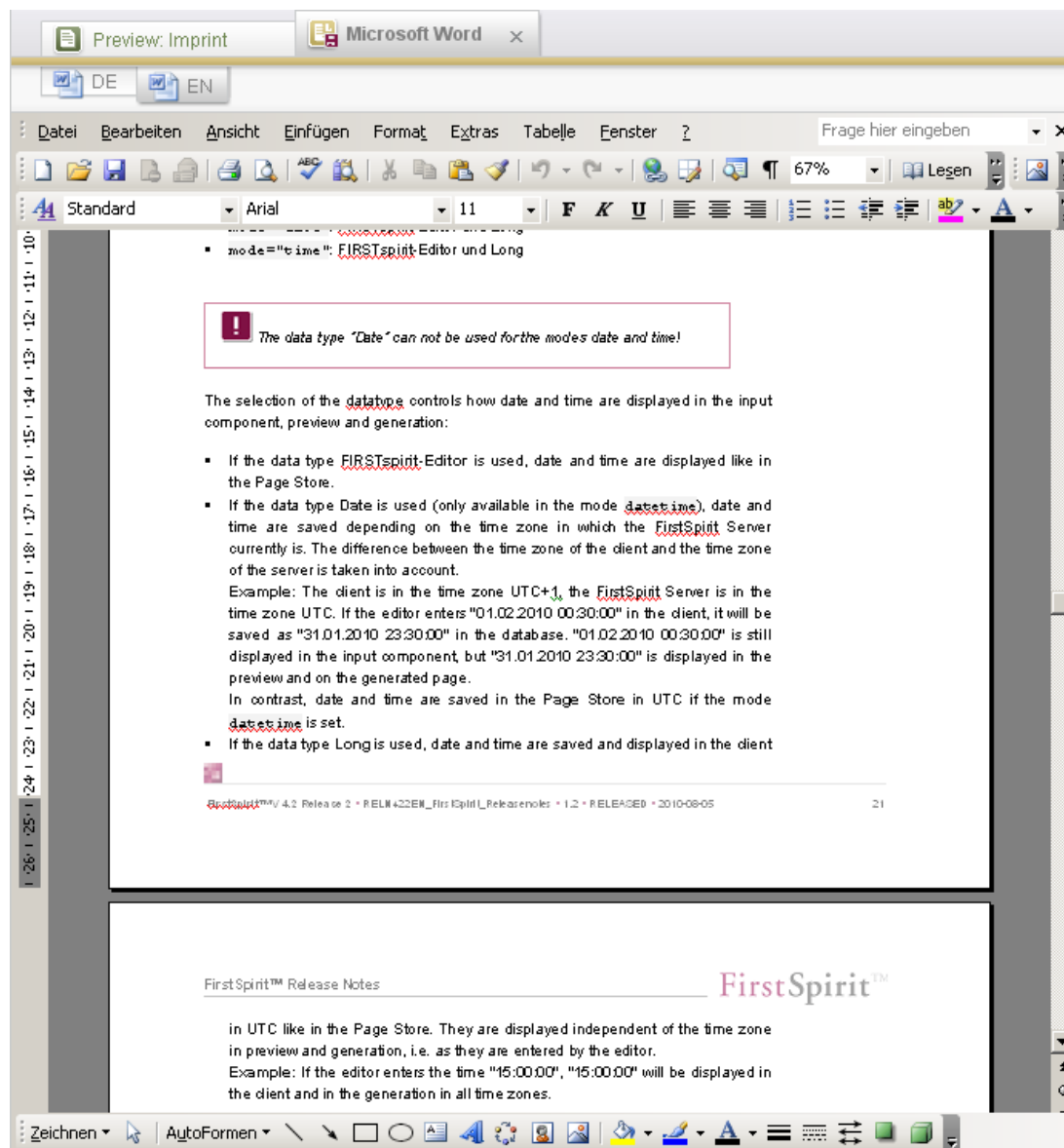
Abbildung 1-13: Display of a Microsoft Word document in the Integrated preview

To this end, several **tabs** in the Integrated preview can be simultaneously opened, e.g. using

- the Preview icon of the JavaClient toolbar,
- Ctrl + P,
- the "Preview" entry of the Context menu

- the new Preview icon on the folder level of the Media Store
- the switch to Edit mode (Lock)

However, the tabs cannot be moved like in the editing area (middle column of

JavaClient): They are always in the order in which they were opened, until they are closed. On the far left-hand side is the tab with the preview of the output channels. The tabs show the name of the respective medium (or "Help", if FirstSpirit Online Help is opened) and the icon for the file type. In addition, the icon is used to show whether it is a language-dependent medium or not and whether the medium has changes which have not been saved.

The synchronisation with the tree structure is constantly active. I.e., if you switch to another tab in the Integrated preview, this medium is switched to in the tree structure and is therefore also displayed in the editing area. If a medium has been opened in the preview from a folder overview, this is displayed by the Folder icon on the respective tab. If this tab is clicked, the folder is displayed in the editing area instead of the medium it contains. A tooltip shows the name of the folder. Open the context menu on the Folder icon to switch to the folder overview.

Further information on the currently displayed file is displayed on the **hanging tabs**:

- "Medium": This is a language-independent medium, a file type icon is also displayed for several file formats.
- "DE" / "EN", etc.: This is a language-dependent medium, a file type or application icon is also displayed for several file formats, it is possible to switch to the Media preview in the desired language via the respective language tab. With PDFs and Flash files the icon of the respective selected browser is displayed.

Close preview / **Close other previews**: These context menu entries are available on each tab, as in the editing area (middle column of the JavaClient).

: This icon can be used to close the individual tabs apart from the one containing the preview of the output channels. If the browser is changed ("View" menu / "Browser engine"), close the tabs also.

> *Other functions related to the Integrated preview cannot be performed for the preview of media and FirstSpirit Online Help. For example, the preview cannot be displayed in a separate window ("View" menu / "Display area of the preview" / "in separate window").*

1.3.2.6   Integrate Preview of media

In the Integrated preview for media, depending on the default settings made by the project administrator, media can be displayed as follows:

- **Images** are displayed in the respective language and different resolutions via a java-based application. To display a certain resolution, the desired resolution must be selected in the resolution list and the Preview icon clicked, or alternatively, activated by means of a double-click.

- **PDF and Flash documents** are displayed with the help of a plug-in of the browser used, with which these documents can be displayed. The relevant plug-in must be installed on the workstation computer of the editor and correctly configured. All the plug-in's functions are available.

- **Office documents** (Microsoft Word, Excel, PowerPoint; OpenOffice Writer, Calc, Impress [BETA test stage only!]) can be opened in the respective Microsoft Office installed on the workstation computer. In this way, a Word document, e.g. can be edited directly in the FirstSpirit Media Store. To do this, it is necessary to switch to Edit mode. Changes must be saved using the Save functions of JavaClient, so that the changed document is available in JavaClient.

> *Using the Integrated Preview for media with Microsoft Office and externally as Microsoft Office desktop application (e.g. started via Windows or in the JavaClient) at the same time, can lead to problems under certain circumstances and is no guaranteed feature of FirstSpirit integration. In this case, either the Integrated preview for Office documents in FirstSpirit **or** the respective Microsoft Office desktop application should be used, no **mixed** mode.*

> *If changes are made to documents, e.g. Microsoft Excel worksheets, which are not in FirstSpirit Edit mode (i.e. are unlocked), information messages of the respective Microsoft Office application can be displayed during the editing, and if applicable can be closed with "OK".*

> *The OpenOffice integration is currently in beta testing stage!*

- **Audio and video files** (e.g. MP3, WMV, AVI, MPG) are played back in the Integrated preview on the Windows Media Player.

- **Text files** (e.g. TXT, HTML, XML, CSS) are displayed via a text editor integrated in FirstSpirit. FirstSpirit syntax is highlighted in colour (syntax highlighting).

> **!** *In order for the required document from the Media Store (e.g. Word file) to be displayed and edited correctly, it is necessary for it to be correctly saved as a picture or file in the FirstSpirit Media Store.*

The media files can also be displayed in a separate window ("View" window / "Display area of the preview" / "in separate window").

2   FirstSpirit start page (Java Web Start)

Initial access to the FirstSpirit server is usually via the Internet. When the start page is opened the system checks whether automatic logon is possible. If a SSO-compatible login module is located on the server and Internet Explorer is simultaneously used, the user is automatically logged on to the FirstSpirit server under their Windows login. If the user is not yet registered on the server under their Windows login they are created as a new external user.

> *If the FirstSpirit server is not accessed via the internet but instead, for example, by means of a command line, parameters for communication between JavaClient and the FirstSpirit server can be configured in a special connection dialog, similar to the connection settings in Chapter 2.2.4.1 from page 42. Further information on the configuration options is given in the FirstSpirit Manual for Administrators, Chapter 6.5.*

> *The simultaneous login via a web browser (e.g. in different windows or tabs) to several FirstSpirit servers with the same host name (e.g. myServer:8200 and myServer:8400) is not supported.*

2.1   Login page

If the automatic login fails (or login via SSO is not configured), a login page appears. A user can log in to the FirstSpirit server via the login window. This login is valid for all applications on the server and is also retained for inactive users for a certain period.

Figure 2-1: Java Web Start – Login

The top of the page contains information on the name and version of the FirstSpirit server.

 This selection box on the top right-hand side of the page can be used to specify a language for further work with FirstSpirit.

> *In addition to the languages English, German, French, Spanish and Russian the menu labels, context menus and dialogs of the FirstSpirit JavaClient are available from FirstSpirit Version 4.1 in Italian and from Version 4.2R2 in Dutch too.*

User: The user's name under which they are logged in to the FirstSpirit server is entered in this field.

Password: The user's password is entered in this field.

> *In rare cases, depending on the authentication method used, only the ASCII characters in passwords are checked. Please contact your server or project administrator for further information..*

Click on the Login button to log in under the entered user name.

2.2 Start page

After logging in (automatically or manually) the FirstSpirit start page opens. The start page is subdivided into areas which are displayed or hidden depending on the permissions of the respective user:

- Quick Start area   (see Chapter 2.2.1)
- Clients area   (see Chapter 2.2.2)
- Administration area   (see Chapter 2.2.3)
- User area   (see Chapter 2.2.4)

The Sun Java Runtime Environment (JRE) Version 1.5.0. or higher is required to start Server and Project Configuration in the Administration area and to start JavaClient, JRE also contains Java Webstart. (The JRE is usually automatically installed when JDK 1.5.0. is installed.)

Figure 2-2: Java Web Start – Start page

The top of the page contains information on the name and version of the FirstSpirit server. In addition, the user currently logged in to the server is also displayed there.

 This selection box on the top right-hand side of the page can be used to specify a language for further work with FirstSpirit.

> *In addition to the languages English, German, French, Spanish and Russian the menu labels, context menus and dialogs of the FirstSpirit JavaClient are available from FirstSpirit Version 4.1 in Italian and from Version 4.2R2 in Dutch too.*

> ![] *The FirstSpirit start page has been reworked, too, in the course of introducing the new "Look & Feel":*

Figure 2-3: New start page

The name of the FirstSpirit server (here "myServer") and the name of the logged-in user (here "Admin") are now displayed in the middle area and down right. The used FirstSpirit version is also displayed down right.

2.2.1 Quick Start area

The left-hand part of the page contains the Quick Start entries which are directly linked to a project. These entries are used to automatically start the client configured for the entry (JavaClient or WebClient) and to open the selected project. Only the projects for which the logged in user has permission to open are displayed in this list.

2.2.2 Client Start area

The entries for starting the FirstSpirit Client are located in the middle part of the page.

- **JavaClient (editing environment):** Click this entry to start the FirstSpirit editing system. The editor can select the required project. A link is automatically established with the server.

- **WebClient (author environment):** Click this entry to start the FirstSpirit author environment via a browser. The FirstSpirit author environment offers a limited scope of functions compared to JavaClient for editorial work with FirstSpirit.

2.2.3 Administration area

The right-hand part of the page is divided again. The top part contains entries for server and/or project administrators.

- **Server monitoring:** Click this entry to open Server Monitoring. A detailed description of this is given in the documentation for administrators.

- **Server and project configuration:** Click this entry to open a console for FirstSpirit Server and Project Configuration. A detailed description of this is given in the documentation for administrators.

> **!** *This area is visible to server and project administrators only.*
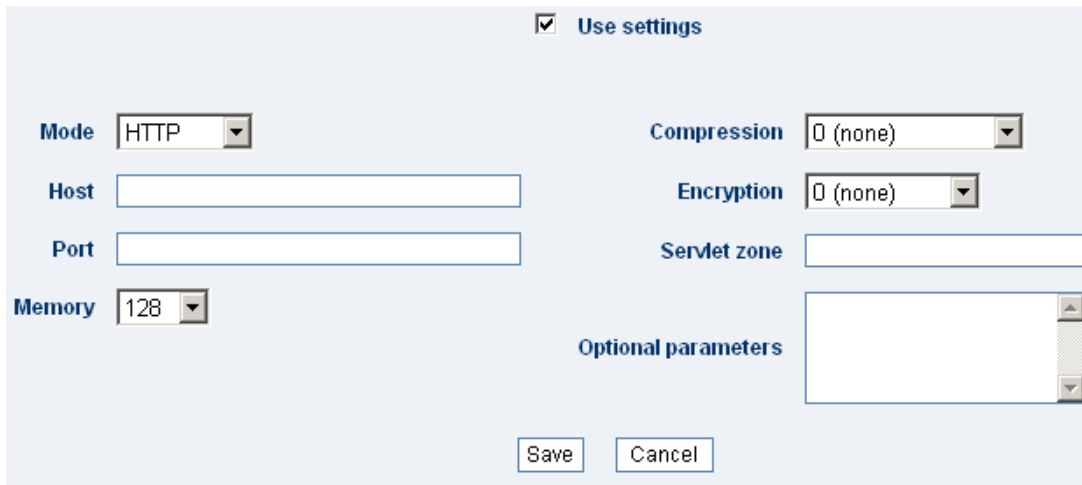
2.2.4 User area

The bottom part of the right-hand side of the window contains the area for the user settings of the currently logged in user:

- **Connection settings:** The connection settings of the currently logged in user can be changed here (see Chapter 2.2.4.1 page 42).

- **Change user:** In several cases the user may want to authenticate themselves under a different user name at the FirstSpirit server, for example to log in as a server administrator (see Chapter 2.2.4.3 page 44).

- **Change password:** The password of the currently logged in user can be changed here (see Chapter 2.2.4.2 page 44).

- **Log out:** Click this entry to quit the current FirstSpirit session for the logged in user (see Chapter 2.2.4.4 page 45).

2.2.4.1 Configuring connection settings

Figure 2-4: Configuring connection settings

Connection settings of the currently logged in user for the start of the JavaClient and the application for Server and Project Configuration.

> *The values configured here overwrite the server-wide valid Webstart settings for this user. The settings should be changed for test purposes only.*

Mode: The connection mode for standard communication between FirstSpirit Clients and the Server can be set for the currently logged in user in the pull-down list:

HTTP: normal Internet connection (default setting)

Socket: direct connection mode.

Host: Server name or IP address of the FirstSpirit server with which the Client is to connect with Web Start.

Port: Port number of the FirstSpirit server.

Memory: The amount of memory (in MB) to be made available for the Client's virtual machine is given here. Valid values are 128m, 256m, 512m or 1024m.

LookAndFeel: Since FirstSpirit version 4.1 the new "Look & Feel" with the name "LightGray" is released for the FirstSpirit JavaClient. From this version on this Look & Feel is the default setting. However, the JavaClient can be used further on with the

classic layout "Classic" without any restrictions. The look can be set by using this combobox.

> ! *The functionality for changing the Look & Feel is only released with FirstSpirit version 4.1.*

Compression: Compression for communication between FirstSpirit clients and the server for the currently logged in user:

None: No compression for the transfer of data between client and server.

Deflate: Use the deflate algorithm with standard compression for the transfer of data between client and server.

Deflate speed: Use the deflate algorithm with fastest compression for the transfer of data between client and server.

Deflate best: Use the deflate algorithm with the best compression for the transfer of data between client and server.

Encryption: Encryption for communication between FirstSpirit clients and the server for the currently logged in user:

None: No encryption for the transfer of data between client and server.

TLS[1]: Use the TLS protocol for the transfer of data between the client and server.

DH ARC4: Use the DH ARC4 encryption algorithm for the transfer of data between the client and server.

Servlet zone: Details of the servlet zone.

Optional parameters: Optional VM parameters for the Webstart configuration can be saved in this field

Click the **Save** button to save the changed connections settings for the currently logged in user. The ☑ **Use settings** checkbox must be activated (selected) to activate the settings. The following information is then displayed on the start page:

Connection settings
ARE ENABLED

---

[1] Transport Layer Security

2.2.4.2   Change password

A user can click this entry to change their password for logging in to the FirstSpirit server.

Password	[ ]

Please enter a new password:

New password	[ ]
Repeat password	[ ]

[Change] [Cancel]

Figure 2-5: Change password

Password: The current password has to be entered again in this field.

New password: The new password is entered in this field.

> [!] *In rare cases, depending on the authentication method used, only the ASCII characters in passwords are checked. Please contact your server or project administrator for further information..*

Repeat password: The new password is entered again in this field to exclude accidental typing errors when changing the password.

Click the [Change] button to adopt the new password for the logged in user.

> [!] *This entry is only available to users manually created on the server. Not for external users created by an automatic SSO login.*

2.2.4.3   Change user

Click this entry to log in another user to the server. The login page opens again (see Chapter 2.1 page 37) which is now extended to include the automatic login option.

Figure 2-6: Java Web Start – Change user

If there is an SSO-compatible login module on the server, on the right-hand side of the screen it is now possible to automatically log in to the server under the Windows login.

Click this button to log in under the Windows login.

2.2.4.4 Log out

By clicking this entry the logged in user can log out from the server. The login page then opens again (see Chapter 2.1 page 37).

2.3 Project selection

After starting the JavaClient the user can now select the project they wish to edit. Only the projects for which the logged in user has the necessary permission appear in the selection list. The permission is assigned by the system administrator. The required project is selected from the selection list and the selection is confirmed with the ☑ button. With the project selection the specific data and settings of the project are then loaded.
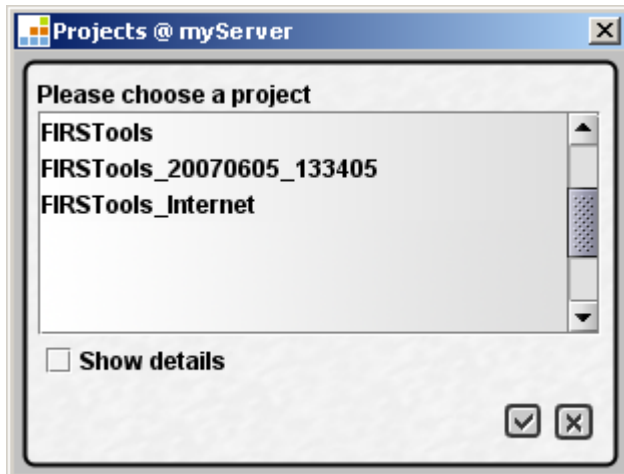
Figure 2-7: Project selection

☑ Show details If the "Display details" checkbox is activated a further dialog window appears with further information on the displayed projects.
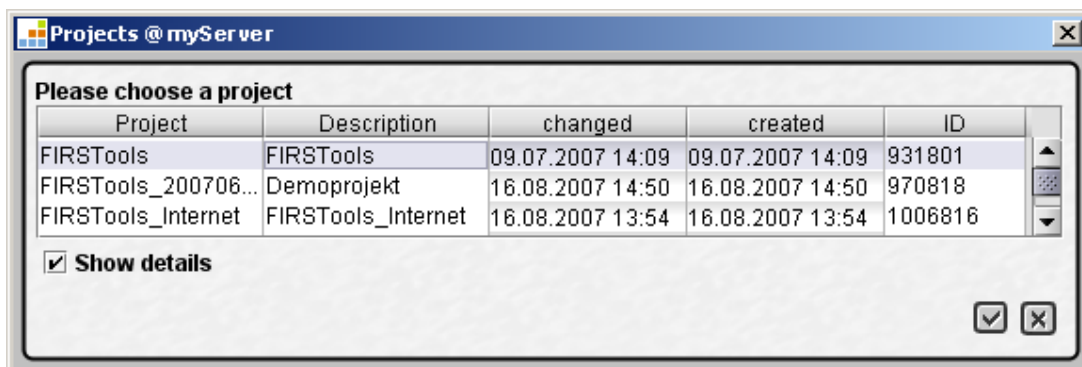
Figure 2-8: Further project details

Confirming a selection with the ☑ button of a selected table cell opens the corresponding project. If the "Display details" checkbox is deactivated the project selection dialog from Figure 2-7 appears again.

After selecting a project the FirstSpirit JavaClient is started.

Figure 2-9: Loading project

From FirstSpirit Version 4.2 the start dialog looks like the following and offers information about the project's loading progress the number of sent and received data, the duration and speed of the starting process so far, the used FirstSpirit version and the server name.

Figure 2-10: Start dialog from FirstSpirit Version 4.2

3   Menus and Icons of the FirstSpirit JavaClient
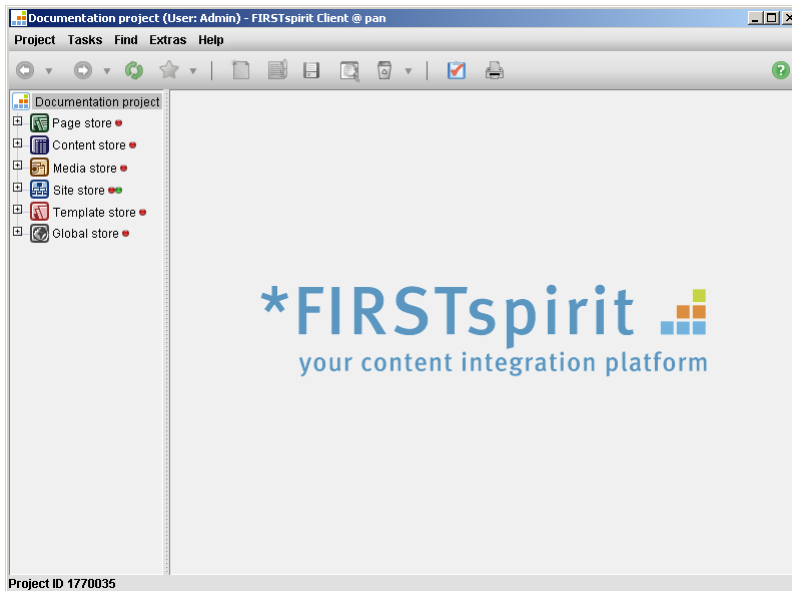
Figure 3-1: Project view in the FirstSpirit JavaClient

Apart from the FirstSpirit logo, the JavaClient **Titlebar** also contains details of the currently opened project (project name), the user name of the logged in user and the server name.

The next line contains the **Menubar**. Explanations on the menu bar are given in Chapter 3.1 page 49. Directly below the menu bar is the **Toolbar** which provides several frequently required functions as buttons. The taskbar functions are described in Chapter 3.2 page 88.

Essentially the screen view of the FirstSpirit JavaClient consists of a left and right-hand side of the window whose sizes relative to each other can be varied. The current project is displayed in the left-hand side in the form of a **tree structure**. Depending on how far the sub-trees are expanded, the folder structure of the six different stores can be seen (from the top down these are: Page, Content, Media, Site, Templates and Global Stores). The right-hand half of ht screen is the **editing area**, which directly refers to the activated position within this tree structure. The currently selected object is highlighted in the tree structure, in the colour of the respective store. After a project is initially opened the FirstSpirit logo is displayed in the editing window on the right-hand side (as can be seen in the screenshot), as no object has been marked yet.

The **status bar** is located in the bottom part of the FirstSpirit JavaClient. Details of

template names, object IDs, etc. are displayed here. The details in the status bar also refer to the object currently selected in the tree structure. From FirstSpirit version 4.2 the following information is displayed here:

- Status (released | not released | in workflow): a dropdown menu containing the next action/s which can be executed for the respective object can be opened with one click werden (see Chapter 12 page 422)
- Version (Date and time of the letzt editing): the version history of the respective object can be opened with one click (see Chapter 11.12.2 page 395)
- Last editor
- in case of pages and sections in the Page Store the used page or section template is shown at the right-hand side. You can jump directly to this template in the Template Store with one click on the template's name.

> *For information about design and functioning of the tree structure and the editing area from FirstSpirit version 4.2 see Chapter 1.3.2 page 25.*

3.1   FirstSpirit menu bar

Project Tasks Find Extras Help

Figure 3-2: Menu bar of the FirstSpirit JavaClient

The FirstSpirit JavaClient makes general and project-wide functions available in a menu bar. All the menu items and sub-menu items are individually explained in the following chapters with a brief description and any limitations.

> *If the functions of individual menu items are not available they are displayed in "grey". Possible reasons for this are: activated or deactivated edit mode, the status of the current object and the permissions assigned to the user.*

The most important functions and most frequently used by users, in the menu bar are also in the toolbar as buttons. The toolbar is located directly below the menu bar and is described in Chapter 3.2 page 88.

3.1.1 Project
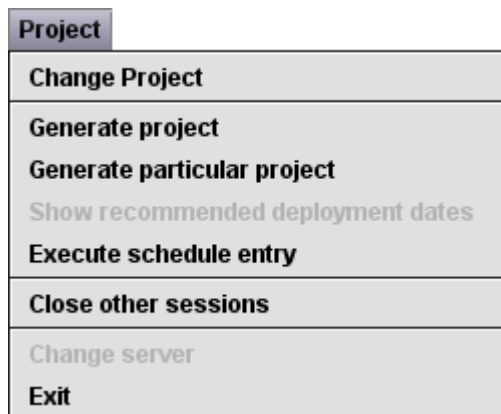
Figure 3-3: Main menu item – Project

3.1.1.1 Change project

This menu function can be used to close the currently open project and to reopen the project selection dialog. A new project can then be selected and opened in the dialog window (see Figure 2-7). Only the projects for which the logged in user has the necessary permission appear in the selection list.

> *A warning is issued if projects in the open project are still in edit mode. If content has been changed in these objects and not yet saved the changes are irrevocably lost when the project is changed! To prevent this, the user can use the search function in the menu bar to leave "objects locked".*

> *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. With FirstSpirit Version 4.2R2 it is available again. However, it is still in a test phase, does not work throughout and can potentially be removed again.*

3.1.1.2 Generate a project

Click this menu entry to generate the project on the server. After generation a log file is displayed with information on the process.

> ⚠️ *The user must have the relevant permission to execute this menu function.*

3.1.1.3 Generate particular project

From time you may only want to generate a subset of a certain project. Such partial generation can be started using the **Generate particular project** function. Which nodes are taken into account depends on the configuration in the project properties (see *FirstSpirit Manual for Administrators*, "Perform Generation" chapter).

If the project administrator has set an option for the user in the project properties, they can select the pages of the Site Store to be generated in a new window. From FirstSpirit Version 4.1, nodes from the Media Store can also be chosen as the starting point. In this way, for example, media not taken into account in a generation can be additionally generated and deployed:

Figure 3-4: Select starting point for partial generation

The +- symbol can be used to further expand the respective Store. Several nodes can be selected by simultaneously pressing the <Ctrl> or <Shift> key. Click **OK** to perform partial generation for all selected starting points (including all lower-level nodes). The address under which the selected starting points are generated is given in a pop-up window; the **Open URL in browser** button can be used to open the address directly in the browser.

> ❗ *In addition to the nodes selected in the **Add Starting Point** dialog, the project administrator can also define other start nodes in the project properties. These are also taken into account in the partial generation. In addition, settings could have been made in the **Project Properties**, which continue to affect generation (see FirstSpirit Manual for Administrators). For example*
> *- the "old" generated status before the generation can be completely deleted on the FirstSpirit server*
> *- the generation can be cancelled, if no changes have been made to the selected nodes since the last generation or if the generation is prevented by media*
> *- only files which have changed are exchanged*

To ensure data integrity is maintained, part-generations are only possible if the Site Store and/or Media Store has been released.

If the editor has not configured any options, the generation starts immediately when the **Generate particular project** function is called. The nodes which the project administrator has defined in the project properties are taken into account for the generation.

3.1.1.4   Show recommended deployment dates

This function is used if validity periods have been defined for individual sections. A list then display when something is to change in the pages to be generated. The user can specify the time at which the pages are to be generated.

> ❗ *The user must have the relevant permission to execute this menu function.*

More details explanations of this topic are given in the FirstSpirit Manual for Administrators.

3.1.1.5   Execute schedule entry

This menu entry can be used to directly execute a deployment. If this menu entry is clicked a dialog window opens with a list of all entries for execution of a deployment for this project which can be started directly.

FirstSpirit Manual for Editors (JavaClient)

First Spirit™

> **!** *The user must have the relevant permission to execute this menu function.*

3.1.1.6   Close other sessions

Each user can work on several projects simultaneously. A further FirstSpirit JavaClient must be started for each additional project.

This menu entry can be used to close all the user's own client sessions, except the active session. A confirmation prompt, whether the sessions should be closed or not.

> **!** *Changes which have not been saved are lost when a session is ended.*

This menu function is also used if the FirstSpirit JavaClient has been accidentally and suddenly quit. In this case the edit mode can remain activated on the server for some time. The edit mode can be cancelled using the "Close other sessions" function.

3.1.1.7   Change server

The current project is ended, after a query. The connection window can now be used to set another server on which the user is to continue working. The client is not closed, i.e. conflicts can arise between the server and client version. Please notify your system administrator if such problems occur.

With this function it is also possible to change the name of the current user without restarting the client. To do this, the user logs in to the current server again and can then log in as a new user by entering a new user name and the corresponding password.

3.1.1.8   Exit

Click this menu item to exit the FirstSpirit JavaClient. A confirmation prompt appears here to prevent accidental exits. In addition, before exiting the user is reminded of any objects still in edit mode. If the changes are not saved they are lost when the JavaClient is exited.

3.1.2 Tasks

Figure 3-5: Main menu item – Tasks

3.1.2.1 Task list

This menu function can be used to insert or display a project-related task list.

Detailed documentation on use of the task list is given in Chapter 11.1, page 314.

The task list can also be opened using the ☑ button in the toolbar.

3.1.2.2 Start workflow (without context)

This menu function can be used to start so-called context-free workflows. Context-free workflows are workflows which are not directly assigned to an object. By contrast, the "Request release" workflow is always assigned to a specific object.

Starting and switching a workflow to another workflow action / state is explained in Chapter 12.2.2 page 425.

3.1.3   Search
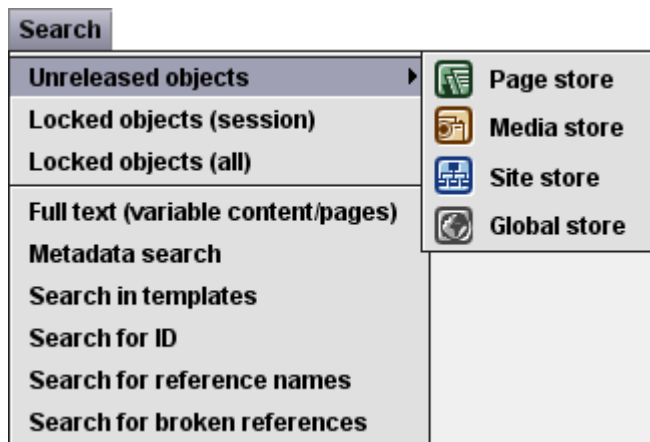
Figure 3-6: Main menu item – Search

- Unreleased objects: see Chapter 3.1.3.1
- Locked objects (session):   see Chapter 3.1.3.2
- Locked objects (all):   see Chapter 3.1.3.3
- Full text (variable content/pages):   see Chapter 3.1.3.4
- Metadata search: see Chapter 3.1.3.5
- Search in templates: see Chapter 3.1.3.6
- Search for ID: see Chapter 3.1.3.7
- Search for reference names: see Chapter 3.1.3.8
- Search for broken references: see Chapter 3.1.3.9
- Search for external references: see Chapter 3.1.3.10

3.1.3.1 Unreleased objects
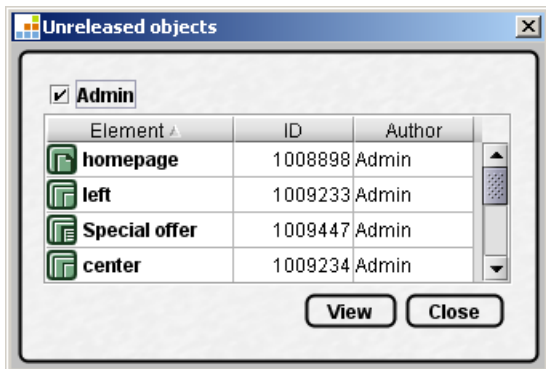
Figure 3-7: List of all unreleased objects

The "Unreleased objects" menu item is subdivided into the individual stores. By clicking a store the user can display all this store's objects which have been changed but have not yet been "released" by a workflow.

☑ **Admin** The name of the logged in user is displayed above the list. If the option is activated before the user name, only the user's own changes are displayed.

View If an object in the selection list is selected (marked), click this button to switch the focus in the JavaClient's edit window directly to the selected object.

3.1.3.2 Locked objects (session)
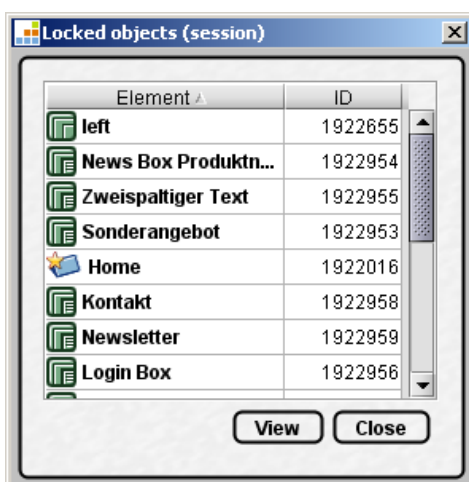
Figure 3-8: List of all locked objects in a session

The user can click the "Locked objects (session)" menu item to display all the objects of a project for which they themselves have activated edit mode.

【 **View** 】 If an object in the selection list is selected, click this button (or double-click the object) to switch the focus in the JavaClient's edit window directly to the selected object.

3.1.3.3   Locked objects (all)
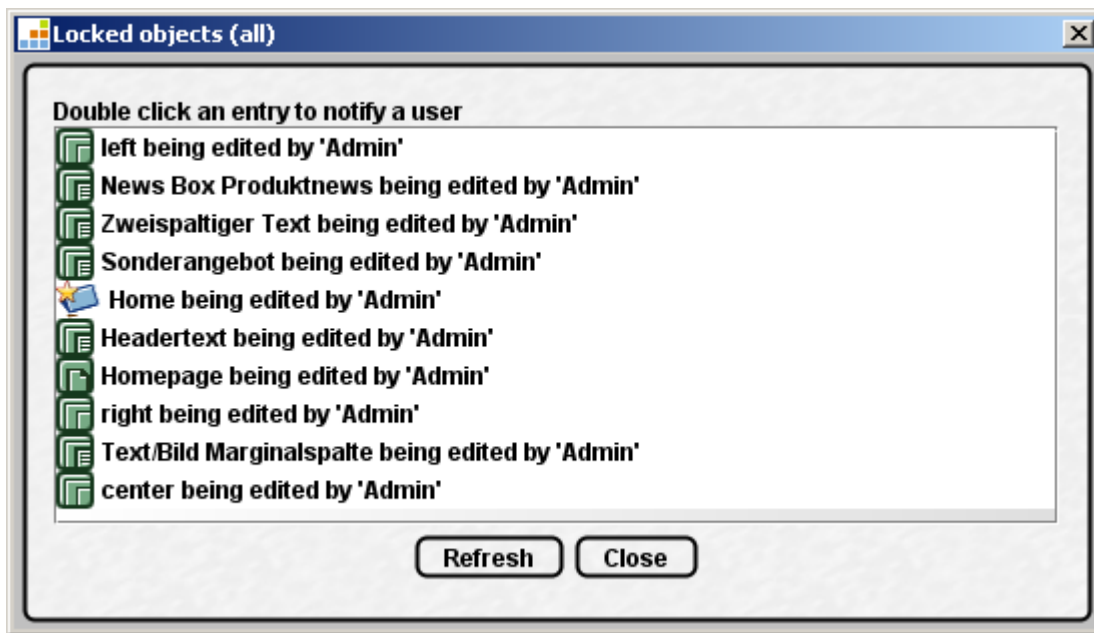
Figure 3-9: List of all locked objects

By clicking the "Locked Objects (all)" the user can display all objects in a project which are currently in edit mode and are therefore protected against access by other users. Apart from the name of the object the list also contains information about which user is currently working with the object.

Double-click a table entry to open the "Send Message" dialog window.
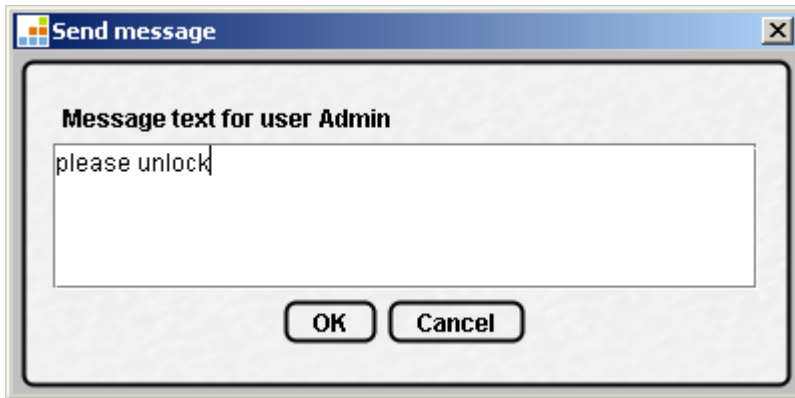
Figure 3-10: Send message

The message is sent to the user currently working on the selected object by clicking the OK button and if necessary asks them to release the object for further editing by other users.

3.1.3.4   Full text (variable content/pages)

This function can be used to browse all stores of the project (exception: database contents) for a specific text. It not only refers to text entries (e.g. in the Page Store) but also to the names of integrated media, corresponding comment lines, the content of media objects (e.g. PDF documents) in the Media Store, etc. The user is informed, in the status bar of the window, whether the search is still running (Searching...) or whether the search has already finished (Search finished!)

The search window is divided into three areas:

▪ Free text search (see Chapter 3.1.3.4.1 page 59)
▪ Editor search (see Chapter 3.1.3.4.2 page 60)
▪ Result (see Chapter 3.1.3.4.3 page 61)

Full text indexing in the Media Store: With version 4.0 documents from the FirstSpirit Media Store are indexed to permit a full text search. Not all document types are supported as a matter of principle and for technical reasons. For this reason, (complete) indexing is not possible in all cases.

Search with wildcards: One or several parts of the words in the search term can be replaced with the * wildcard.

Search text	Result (examples)
Motor*	Motor, Motorway, Motorway junction
*way	Carriageway, Motorway
M*way*	Motorway, Motorway junction

3.1.3.4.1   Free text search

The "Free Text Search" tab can be used to search for text fragments located in any object within the project.
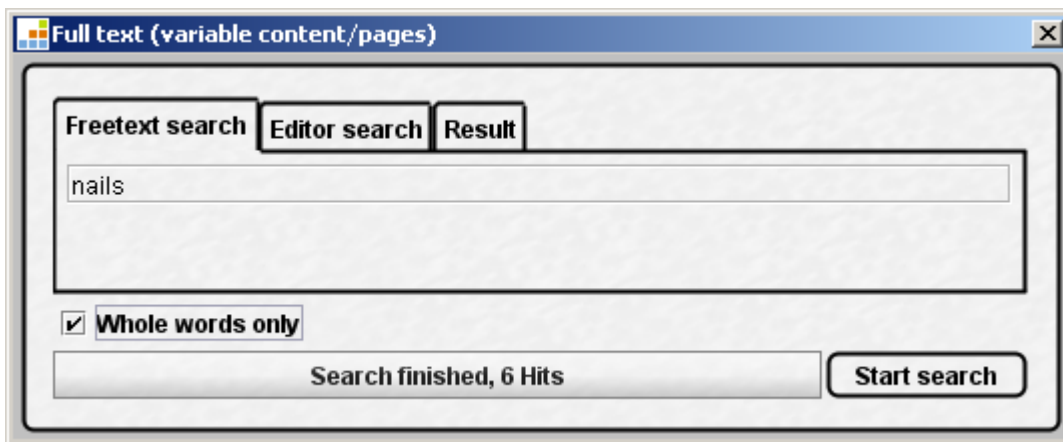
Figure 3-11: Full text search (free text)

Whole words only: If this option is activated the search only supplies results with exactly the same notation. Otherwise a partial match also suffices. The following table contains examples of the effects of this option on the result:

Search text	Result (activated)	Result (deactivated)
Motor	Motor	Motor, Motorway
Motorway	Motorway	Motor, Motorway

Start search Click this button to start the search. A progress bar shows you how many elements have already been searched. The search can be interrupted at any time by clicking this button again.

3.1.3.4.2   Editor search

The "Editor search" tab can be used to search in specific input components within the project for text fragments or specific contents.
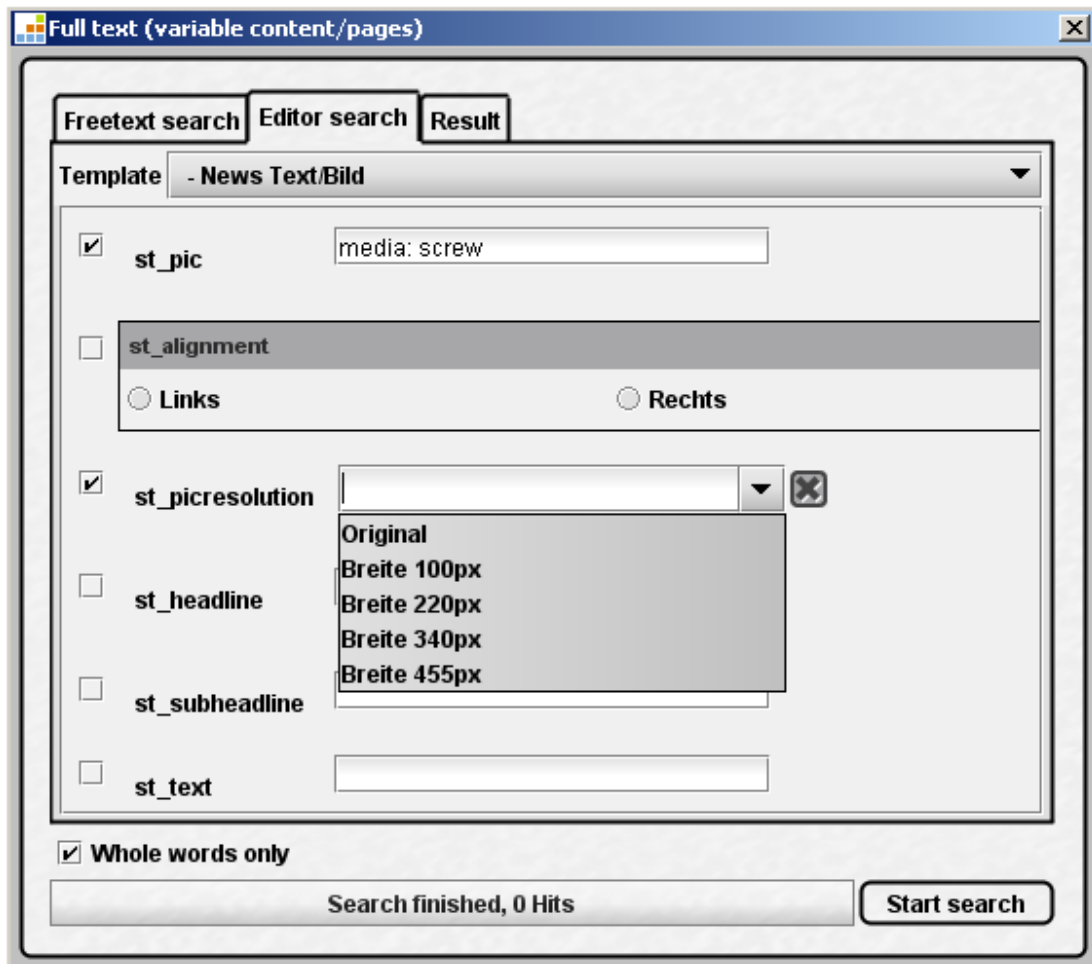
Figure 3-12: Full text search (Editor Search)

Template: With this combo box the user can select the required page or section template to be searched for specific content. A list of all the selected template's input components appears.

The checkbox in front of the input components in which the required content is to be searched must be selected. The search text must be entered in the respective text field of the selected input components.

If more than one input component is required at the same time, all contents must match the search criteria (AND relation).

For the input components **Single line text entry**, **DOM Editor, DOM Table** the

function is precisely the same as for free text search.

For the **Picture Selection** input component it is possible to enter text which is searched for in the reference line of the component. E.g. if the word "media" is searched for without the option "Whole Words Only", all metadata elements in which an picture has been defined are displayed.

For the **combo box** input component a specific entry from the list can be selected and this is then the search term.

3.1.3.4.3 Result

All elements matching the given search criteria are listed in the "Result" tab.
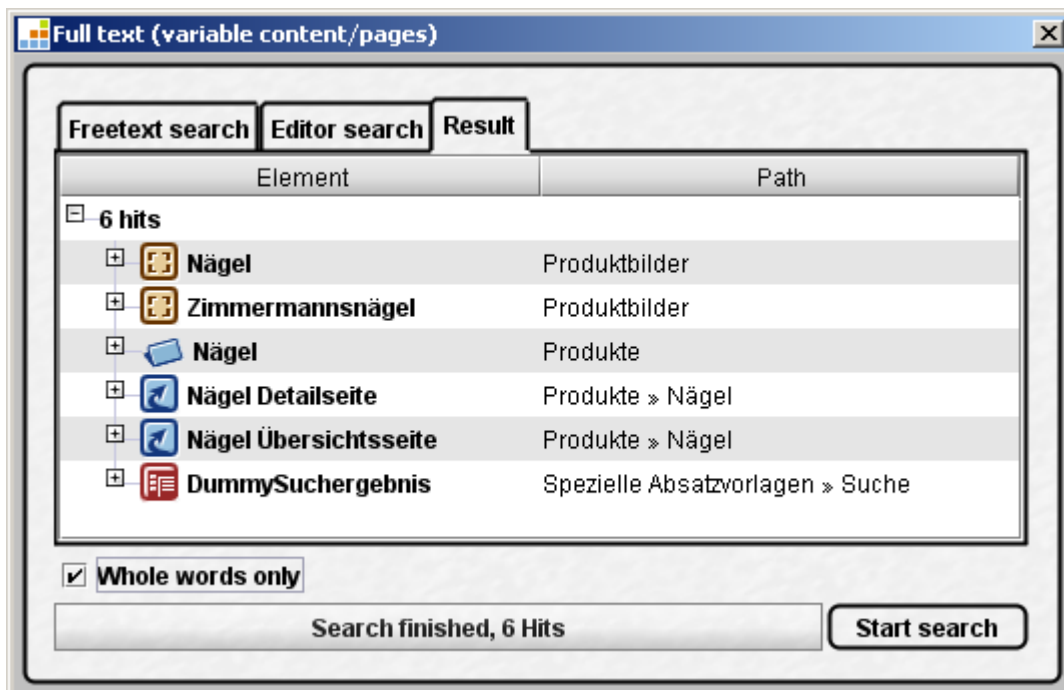
Figure 3-13: Free text search results

Double-click an object in the results list to switch the focus in the tree structure directly to the selected object.

3.1.3.5 Metadata search

This function can be used to search for a specific text in the metadata areas of the project. The search window is divided into three areas:

- Free text search (see Chapter 3.1.3.5.1 page 62)
- Editor search (see Chapter 3.1.3.5.2 page 62)

- Result (see Chapter 3.1.3.5.3 page 63)

Search with wildcards: One or several parts of the words in the search term can be replaced with the * wildcard.

Search text	Result (examples)
Motor*	Motor, Motorway, Motorway junction
*way	Carriageway, Motorway
M*way*	Motorway, Motorway junction

3.1.3.5.1 Free text search

The "Free Text Search" tab can be used to search for text fragments located in any object within the metadata.

Free text search in the metadata is analogous to the free text search in the full text search (see Chapter 3.1.3.4.1 page 59).

3.1.3.5.2 Editor search

The "Editor search" tab can be used to search in specific input components within the metadata for text fragments or specific content.
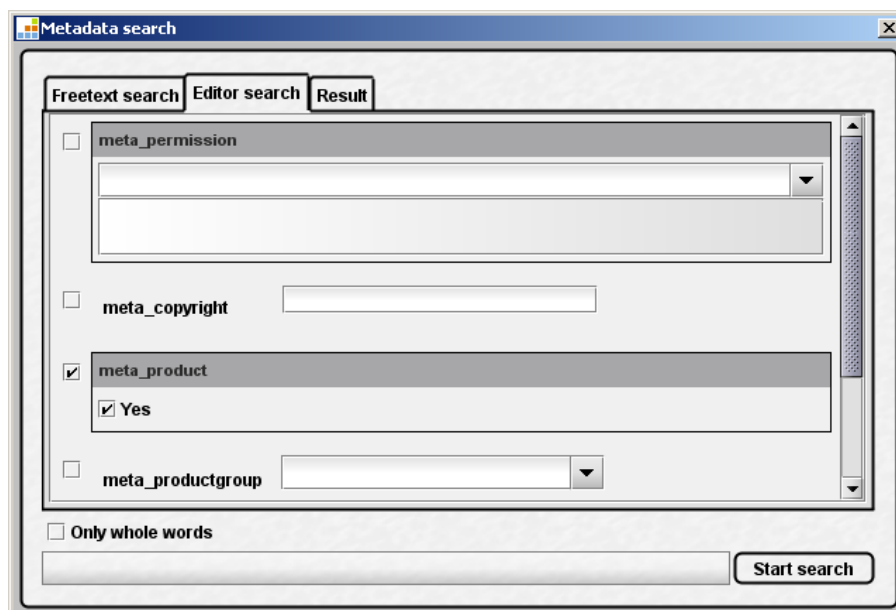
Figure 3-14: Editor search for metadata

The search form depends on the metadata template used in the project.

Editor Search in the metadata is analogous to Editor Search in the Full Text Search (see Chapter 3.1.3.4.2 page 60).

3.1.3.5.3   Result

All metadata elements matching the given search criteria are listed in the "Result" tab.

The result of the Metadata Search is analogous to the Full Text Search (see Chapter 3.1.3.4.3 page 61).

3.1.3.6   Search in templates

This function can be used to search the Template Store of the project for specific content.
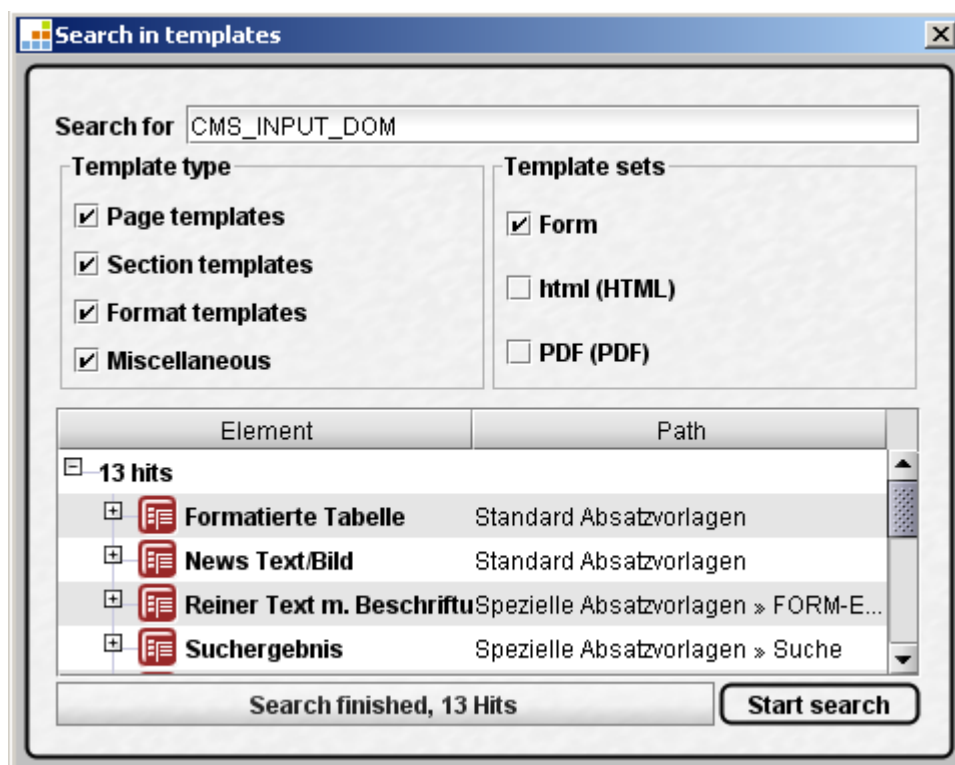
Figure 3-15: Search for Templates

Search for: The text to be searched for in the selected templates is entered in this field. Part of a word can be replaced by the * wildcard.

Search with wildcards: One or several parts of the words in the search term can be replaced with the * wildcard.

Search text	Result (examples)
Motor*	Motor, Motorway, Motorway junction
*way	Carriageway, Motorway
M*way*	Motorway, Motorway junction

Template type: Here it is possible to select the areas of the Template Store in which the search is to be carried out. If the tick in front of several template types is deactivated these template types are excluded from the search. The search function will therefore display results quicker.

Template sets: Here it is possible to select the tabs of a template in which the search is to be carried out. If the tick in front of several template sets is deactivated these template sets are excluded from the search. The search function will therefore display results quicker.

Search results: The results of the search are listed in the area below the search criteria, in the same window. All elements which satisfy the search criteria are listed here.

Within the search results the templates found can be expanded and contracted analogous to the tree display. Double-click a search result to switch the focus directly to the corresponding element in the tree view. The tab in which the search term was found is displayed in the client's right-hand editing area.

Whole words only: If this option is activated the search only supplies results with exactly the same notation. Otherwise a partial match also suffices. The following table contains examples of the effects of this option on the result:

Start search Click this button to start the search. A progress bar shows you how many elements have already been searched. The search can be interrupted at any time by clicking this button again.

3.1.3.7 Search for ID

Each object in FirstSpirit has a unique identification number (ID). As these numbers are **automatically** assigned by the system, it is possible to know an ID but not to

know which specific object it relates to. This function can be used to search for the object corresponding to an ID number.
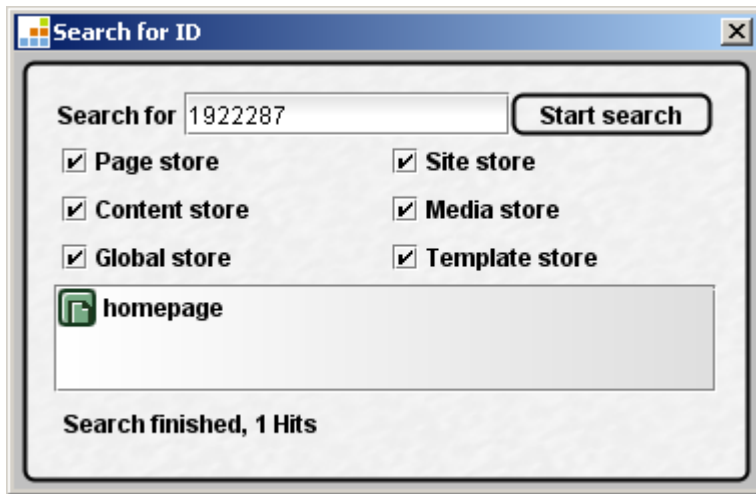
Figure 3-16: Search for ID

Search for: The known ID is entered in this field. Click the [Start search] button to search for the ID number in the selected stores.

The object with its assigned reference name is displayed in the bottom area. Click the object in the results list to switch the focus in the tree structure directly to the selected object.

If the tick in front of several stores is deactivated these stores are excluded from the search. The search function will therefore display results quicker.

3.1.3.8   Search for reference names

Each object not only has its own ID (see above), but also a reference name which must be unique for each store. Each object can be identified by its reference name. This function enables searches for reference names.

> ⚠️ *The reference names of an object and the language-dependent name of the object which, for example, is displayed in the project's tree view can differ. Reference names are only shown in the project if the "Reference Names" setting has been activated in the "Extras / Tree Display" menu, from FirstSpirit Version 4.2 in the "View / Preferred display language" menu (see Chapter 3.1.4.2 page 69).*
>
> *Reference names can also be accessed by the keyboard shortcut **ALT + P** (see Figure 3-21, page 70).*
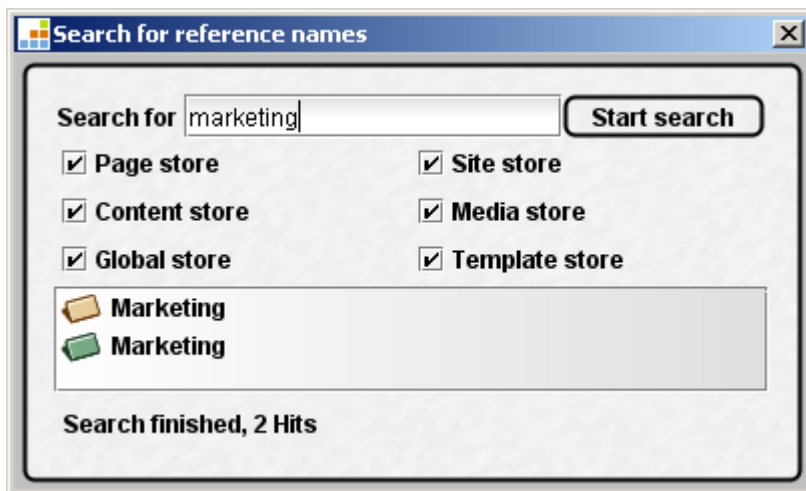
Figure 3-17: Search for reference names

The search for reference names is performed analogous to the search for ID (see Chapter 3.1.3.6 page 63).

Search with wildcards: One or several parts of the words in the search term can be replaced with the * wildcard.

Search text	Result (examples)
Motor*	Motor, Motorway, Motorway junction
*way	Carriageway, Motorway
M*way*	Motorway, Motorway junction

3.1.3.9   Search for broken references
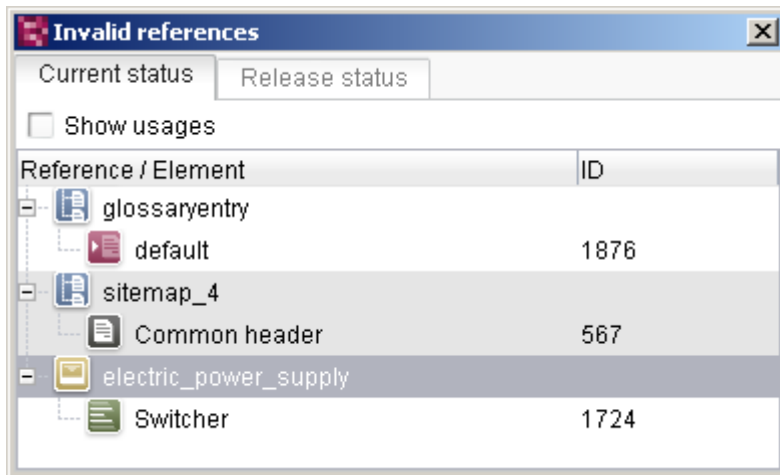
Figure 3-18: Search for broken references

Deleting elements still referenced within the project (or by a defective project import) can cause broken references in the project. This can have negative effects, for example is a reference to a page which no longer exists is opened using the navigation. As such reference errors are difficult to uncover, the search function can be used to directly search for such defective places within the project. Click the menu entry to open the "Broken References" dialog (see Figure 3-18).

All broken references in the project are displayed in the overview. According to the project status they are divided into broken references in the current (i.e. not released) project status and broken references in the release status. The references are displayed on different tabs, according to their specific status. As the elements no longer exist in the project in many cases only the former ID (within the project) can be displayed. It is possible to tell which type of reference it is from the icons.

☑ Show usages The uses of each broken reference in the project can be displayed by selecting the checkbox. The uses are displayed below the reference in the display. Double-click the use to switch the focus in JavaClient directly to the corresponding element to revise the broken reference.

> ■ *If input components contain invalid references to media, files or pages, they are flagged with a red bar, invalid page references are highlighted in red:*
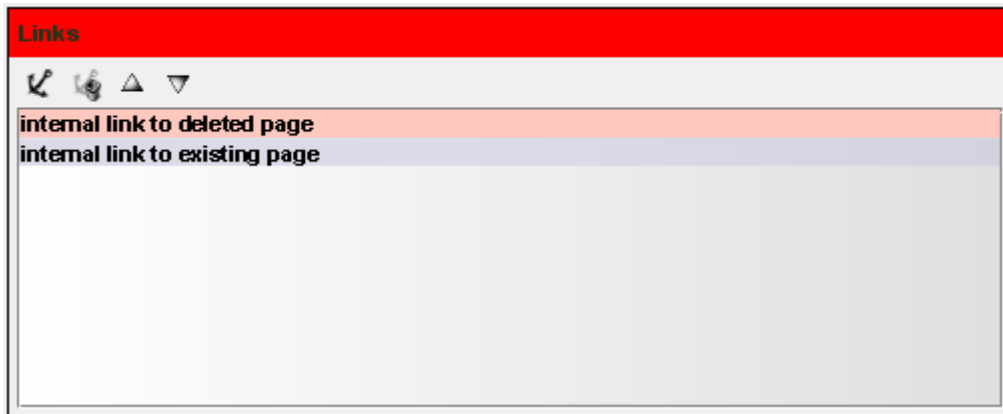
Figure 3-19: Input component with invalid internal link

3.1.3.10 Search for external references

Figure 3-20: Search for external references

This menu item can be used to list all external references (e.g. references to external websites), used in the project. In this way, external links can be quickly determined and for example checked to see how up-to-date they are. All references in the project are displayed in the overview with the display name and idea of the section in which the reference is used. In the case of references in datasets from the Content Store, the reference name of the content (data source) and the ID of the dataset are displayed. It is possible to tell which type of reference it is from the icons. From FirstSpirit Version 4.2, a category assigned by the template developer can be displayed behind the display name, e.g. "url" for internet addresses or "email" for e-mail addresses, depending on which link template has been used for the external link.

☑ Show usages The uses of each external reference in the project can be displayed by selecting the checkbox. The uses are displayed below the reference in the display. Double-click the use to switch the focus in JavaClient directly to the corresponding element.

3.1.4 View (V4.2 and higher)

> **!** *This menu item has been added in FirstSpirit Version 4.2. Einige Unter-Menüpunkte wurden aus dem Menüpunkt "Extras" übernommen.*

3.1.4.1 Show symbols (Metadata, Packages, Permissions)

If this menu function is activated, all nodes to which permissions have been assigned are marked by the icon 👤 or a red dot (Look & Feel "Classic"). In addition, all nodes for which metadata is defined are denoted by the icon ℹ or a green dot (Look & Feel "Classic"). Objects managed with the FirstSpirit Package Management function are denoted by the icon ▦ or a blue dot (Look & Feel "Classic") (further information of FirstSpirit Package Management is given in the documentation "PackagePool").

3.1.4.2 Preferred display language

This function can be used above all in multilingual projects. In the upper area of the context menu the radio buttons can be used to select the preferred display language. All languages which are defined by the project administrator as editorial languages are displayed here.

When the preferred display language is selected the entries in the tree will switch to the respective language-dependent display name and language-dependent contents deriving from the templates, e.g. the labels of the forms, will switch to the preferred display language as well (in case they are provided in the template by the template developer).

> **!** *This setting does not have an effect on the language setting of the FirstSpirit clients. The labels for the tool bar or dialogs etc. are determined by the language settings on the Start page (locale language) (see Chapter 2.2 page 39).*

Always display forms in the "preferred display language": If this menu function is activated, language-dependent content from the template is displayed in the preferred display language (provided it was defined by the template developer in the form area). For example, this concerns the labelling of the Page Store's input components (e.g. text input fields). If this menu function is deactivated, the input components are labelled in the respective project language (content language). In this case, the labelling is therefore dependent on the selected language tab. If the content is not available (in the template) in the required language or the content language, the language information of the default display language is adopted.

Display reference names in the tree: If this checkbox is activated, the reference names of all objects in the tree structure are displayed. If this checkbox is deactivated, the objects in the tree structure are displayed in the preferred display language. From FirstSpirit Version 4.2R2 this setting will be saved for each user and used for the user until they change it.

For information on multilingualism in FirstSpirit projects, see also Chapter 1.2.3 page 20.

> ! *Reference names can also be called using the keyboard shortcut "**ALT + P**". To do this, the respective object must be selected (marked). The following pop-up opens:*
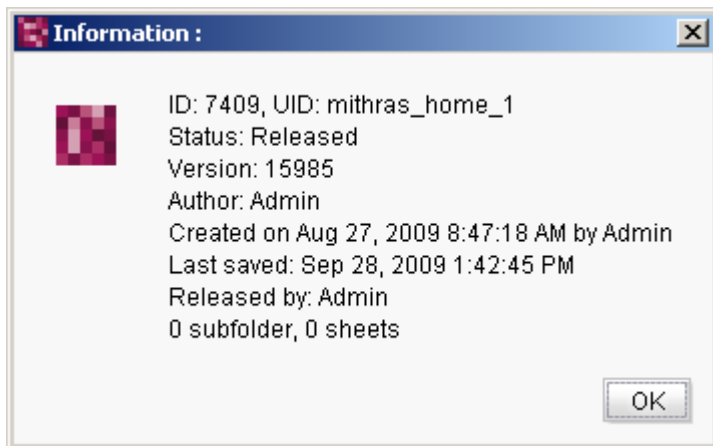
Figure 3-21: Node informations (ALT + P)

3.1.4.3   Workspaces (V4.2 and higher)

Working with workspaces (see Chapter 1.3.2.1 page 25) can be configured globally

by the editor by using this menu item.

Open new in background: If this option is activated new workspaces are opened in the background. If it is disabled new workspaces can be fetched directly into the foreground as an active workspace.

Synchronise new with tree: If this option is activated, 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.
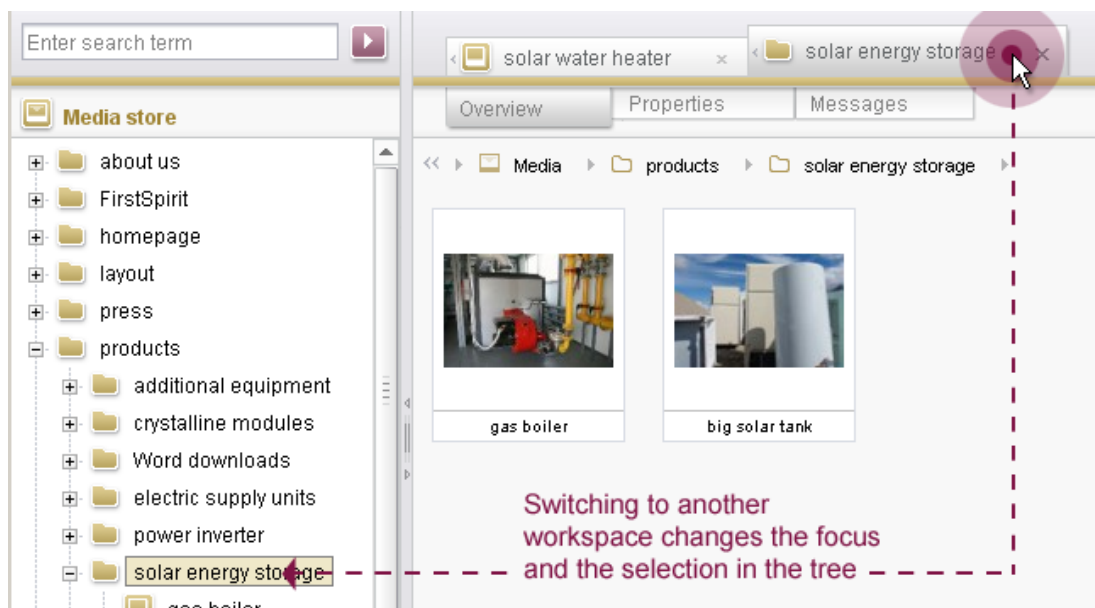
Figure 3-22: Tree synchronisation on changing a workspace

If the option is disabled, there will be no synchronisation.

This option can also be set for single tabs (see Chapter 1.3.2.1 page 25).

Save on Exiting: If this option is activated the user specific workspace (including all opened tabs) will be saved when closing the JavaClient and restored at the next start of the JavaClient. For this purpose, the option "Workspaces" must be activated in the menu "View" / "Restore settings on restart" additionally.

Save open workspaces: This option is used for saving the user specific workspace (including all opened tabs).

Restore closed workspaces: This option is used for reopening workspaces which have been closed. The last ten closed workspaces of the current user session are shown. If no workspaces have been closed during the current session yet, the notice

"not available" is displayed.

3.1.4.4   Display integrated preview (V4.2 and higher)

Using this menu item the integrated preview (see Chapter 1.3.2.4 page 30) can be activated. If the option is activated the integrated preview can be configured using the menu item "Display area of the preview" (see Chapter 3.1.4.5 page 72).

In FirstSpirit Version 4.2R4 and higher, the entries "use for content" and "use for media" are available under this menu item. If the "**use for content**" option is selected, the Integrated preview is used as it was up to and including Version 4.2R2 (see also chapter 1.3.2.4 from page 30), i.e. for content from the Page Store and the Site Store. If this option is deactivated, the "Content Highlighting control" (see chapter 3.1.4.6 page 72) and "Browser engine" (see chapter 3.1.4.9 page 76) menu items are also deactivated. If the "**use for media**" option is selected, the Integrated preview is also used for media (Media Store) (see also chapter 1.3.2.5 from page 32). In this case, the menu item "Office engine" (see chapter 3.1.4.10 page 76) becomes active.

> *Depending on the configuration by the project administrator in the project properties this menu item can be inactive.*

3.1.4.5   Display area of the preview (V4.2 and higher)

If the integrated preview is activated (see Chapter 3.1.4.4 page 72) it 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 3.1.4.6).

3.1.4.6   Content highlighting control (V4.2 and higher)

The "**Content Highlighting**" function provides further help to the editor, apart from integrated preview. 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 of section is outlined with a frame.

"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.

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.

- **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.

▪ **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.

The Content Highlighting is configured using the menu item "Content highlighting control":

Figure 3-23: "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.

▪ **Disable:** Content Highlighting will be disabled.

> ! *Depending on the configuration by the project administrator in the project properties this menu item can be inactive and one of these options can be preset definitely.*

In addition, it is also possible to enable "Content Highlighting" for an external web server (**"Workspace Follows External Web Browser"** (see Chapter 3.1.4.7 page 75). 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.

> ❗ *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.*
>
> ❗ *FirstSpirit components do not all completely support the "Content Highlighting" function. This is especially true for multi-valued components, for example, CMS_INPUT_CONTENTAREALIST.*
>
> ❗ *Content Highlighting is not supported for display of the integrated preview in an external window (cf. Chapter 1.3.2.4 page 30).*

Further information about the Content highlighting see FirstSpirit Online Documentation, Chapter "Advanced topics" / "Content highlighting ".

3.1.4.7   Workspace follows external web browser (V4.2 and higher)

If this menu item is activated and the integrated preview (see Chapter 3.1.4.4 page 72) is deactivated, when navigating through the preview the related page of the Page Store is always activated automatically in the tree structure of the JavaClient. In this way, content of the pages can be edited quick and easy.

3.1.4.8   Restore settings on restart (V4.2 and higher)

This menu item can 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.

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 Chapter 3.1.4.3 page 70). These workspaces can be automatically re-opened the next time JavaClient is started.

> **!** *If the "Workspaces" option is activated, the more workspaces are open on exiting the longer it takes to restart the project in JavaClient.*

3.1.4.9 Browser engine (V4.2R2 and higher)

Use this menu item to choose if the Mozilla Firefox or the Microsoft Internet Explorer is to be used for the integrated preview on Windows platforms. The selected browser is saved in the user profile. JExplorer provides the same functions as the JXBrowser (see Chapter 1.3.2.4 page 30). The selected browser is indicated in the bottom right-hand corner of the integrated preview.

Unlike use of the JXBrowser, which is included in FirstSpirit from Version 4.2, if the JExplorer is selected, the locally installed Microsoft Internet Explorer of the operating system is used. The user-specific configuration is therefore also used.

Versions 6, 7 and 8 of Microsoft Internet Explorer are supported for use with the integrated preview, however only in 32 bit environments and not under MacOS X and Linux.

3.1.4.10 Office engine (V4.2R4 and higher)

If the "integrated preview - use for media" option is enabled, this menu can be used to set which application is to be used for Microsoft Office and OpenOffice file formats. The respective application must be installed on the workstation.

- **Microsoft Office (Windows only):** If this option is enabled, the relevant Microsoft Office application is used to display und edit office documents in the Integrated preview. This menu item is not displayed if JavaClient is operated in an operating system other than Windows (see Chapter 6.7.4 page 199).
- **OpenOffice (BETA, not MacOS):** If this option is enabled, the relevant OpenOffice application is used to display and edit office documents in the Integrated preview. This menu item is not displayed if JavaClient is operated for Mac OS (see Chapter 6.7.5 page 200).
- **Google Docs (BETA):** If this option is enabled, Google Docs is used to display and edit office documents in the Integrated preview.
- **disabled:** If this option is enabled, neither Microsoft Office nor OpenOffice is used to display office documents; instead, they continue to be opened in an external application as to date.

> [!] *If a suitable application for editing the file is not installed on the workstation, it is not possible to edit via the Integrated preview.*

> [!] *If using applications in the Integrated preview, please note that FirstSpirit provides the interfaces required for the application integration, but in general does not have any influence on the integrated applications themselves. Integrated external applications are not part of the FirstSpirit product. Among other things, this means that responsibility for the function of the integrated applications lies with the manufacturer of the application or with the customer or partner who implements the application. (See also FirstSpirit Release Notes Version 4.2R4, chapter 3 "The FirstSpirit AppCenter"))*

3.1.4.11 Graphic-engine (V4.2R4 and higher)

If the option "integrated preview - use for media" is activated, you can decide via this menu, which application is to be used for which picture file format.

- **Java Image Editor:** If this option is activated, the Java Image Editor is used for displaying and editing pictures (see Chapter 6.7.1 page 190).
- **Simple image processing (Picnik):** If this option is activated, the online image processing service www.picnik.de is used for displaying and editing pictures. Picnik makes a simple, intuitive alteration of images. (see Chapter 6.7.2 page 197).
- **Enhanced image processing (Pixlr):** If this option is activated, the online image processing service www.pixlr.com is used for displaying and editing pictures. Modeled more professional image editing software like Adobe Photoshop (see Chapter 6.7.3 page 198).
- **disabled:** If this option is activated, the image processing functions, well-known in FirstSpirit, are available.

> [!] *Use of the **Picnik** and **Pixlr** editors requires an active internet connection. Further information is given on the corresponding internet sites.*

> **!** *If using applications in the Integrated preview, please note that FirstSpirit provides the interfaces required for the application integration, but in general does not have any influence on the integrated applications themselves. Integrated external applications are not part of the FirstSpirit product. Among other things, this means that responsibility for the function of the integrated applications lies with the manufacturer of the application or with the customer or partner who implements the application. (See also FirstSpirit Release Notes Version 4.2R4, chapter 3 "The FirstSpirit AppCenter")*

3.1.5 Extras

> **!** *Some sub menu items which were located beneath the menu item "Extras" up to FirstSpirit version 4.1, have been adopted to the menu item "View" in FirstSpirit version 4.2 (see Chapter 3.1.4 page 69).*

3.1.5.1 Open tasklist on startup (up to and including V4.1)

If this menu function is activated the task list with the pending tasks is automatically opened the next time the FirstSpirit JavaClient is started.

3.1.5.2 Confirm Move operations

> **!** *This function is released from FirstSpirit version 4.1.*

This menu function is activated by default. When objects are moved by "Drag & Drop" the following question is displayed:

Figure 3-24: Confirm move operations

In this way, moving inadvertently some elements within the project is prevented. By deactivating this menu function the question can be switched off.

If folder or pages are moved from the Page Store to the Site Store for creating new menu levels or page references (see Chapter 7.2.1.1 page 217), this confirmation dialog is not displayed.

3.1.5.3   Display preview errors

If this menu function is activated all possible errors are automatically displayed when a preview is rendered.

3.1.5.4   Display preview warnings

If this menu function is activated all warnings are also displayed when a preview is rendered.

3.1.5.5   Follow browser navigation (up to V4.1 including)

If this menu function is activated the corresponding page of the Page Store is always automatically activated in the tree structure of the JavaClient when navigating through the preview. Changes to the content of a page can therefore be made quickly and easily.

3.1.5.6   LookAndFeel (only V4.1)

Using this function the view of the graphical user interface can be changed (see Chapter 1.3 page 23). The change will only become visible when the JavaClient is restarted using the start page and after having logged-in anew.

3.1.5.7 Translation help

If content has already been added in a language, this menu function can be used to easily copy the existing content into other languages. The translation help can be used in the Page Store and in the Content Store

Click the "Translation help" entry to open a dialog with which the content of the language A input components / table columns can be copied into the input components / table columns of language B.
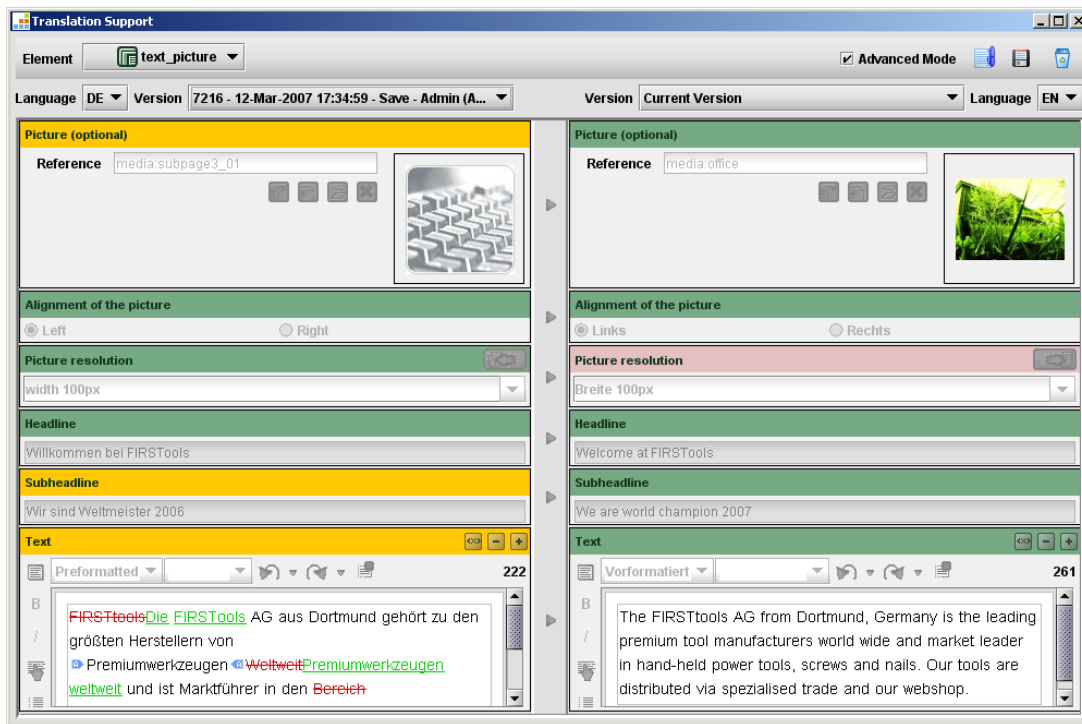
Figure 3-25: Translation assistant – Page Store

The object selected for the translation aid is displayed in the tree view in the **top/middle of the window**. All sections of a page and the page itself can be selected here. The input components of the selected object are displayed directly in the View or Edit window.

Edit mode: The drop-down list can be used to select whether the content is to be completely translated, compared with each other only or if individual parts are to be translated.

The **bottom/left window area** is the view area of a language A. The language-dependent content for translation can be copied from this area into the input components of another language. It is not possible to edit the content; this protects the original language from accidental changes during the translation. The language

tabs can also be used to select a different language to the original language.

The **bottom/right window area** is the editing area of a language B. The contents of the input components from the left-hand side of the window (view area of language A) can be copied into this area. This copied content can then be edited. Another language can also be selected using the language tabs.

From FirstSpirit V4.2R4 the dialog for translation help has been visually revised
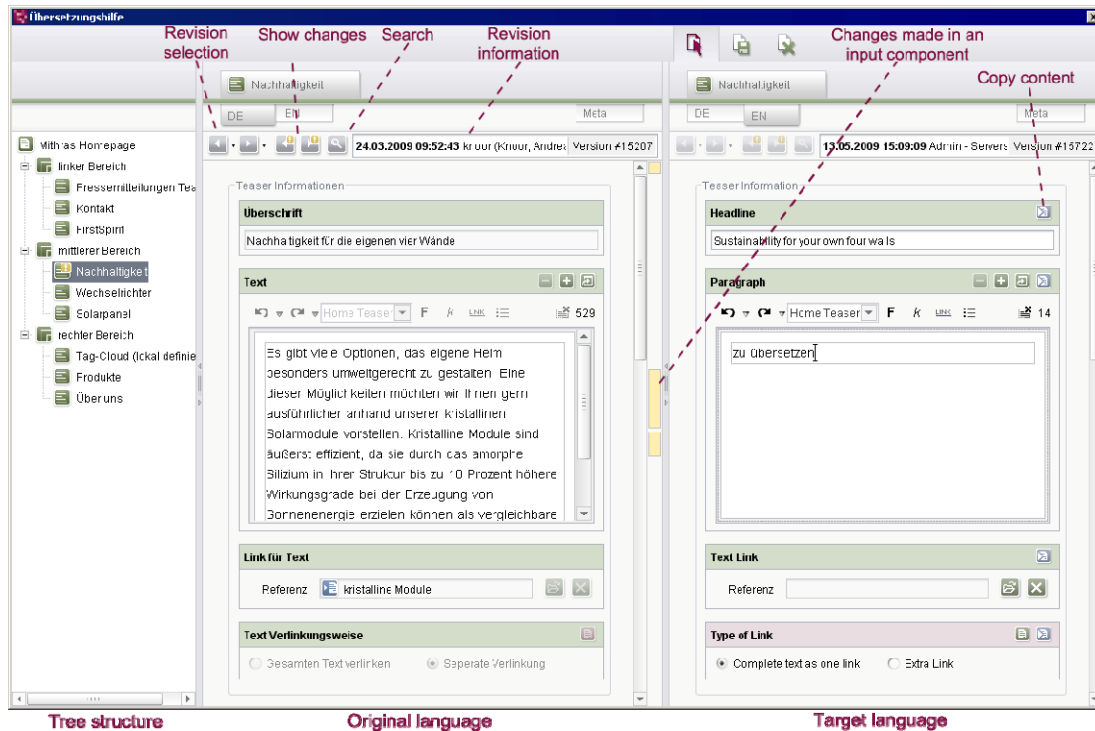
Figure 3-26: Translation help in Version 4.2R4

The tree structure is now shown in a separate window on the left-hand side, as familiar from other dialogs. The input components of the selected object are displayed in two other areas of the window, each in the available languages. For translations, the original language is displayed on the left and the target language in the last revision of the object is displayed on the right. The language tabs can be used to also select languages other than the source or target language. To protect the original language from inadvertent changes, it is not possible to edit the content.

With help of the ⌧ icon can be copied from the original language in the respective input component. The translation can then be made in this dialog, or as usual in the section concerned or the page concerned in the working space of the JavaClient. Each time "Save" is pressed (or Ctrl + S) or the "Switch to View mode" icon (Ctrl + E), a new revision is generated with the comment "Translated". Original language

content cannot be edited.

This dialog can also make changes to older versions are displayed. Between the window areas, a bar shows which input components contain changes. Here, yellow marking indicates that data has been changed in the input component, red marking indicates that data has been removed and green marking that values have been added. A tooltip shows the name of the relevant input component and it is possible to jump directly to the relevant input component with a click.

Icons are used to visualise in the tree structure whether and what type of differences exist between the compared versions. E.g. a yellow exclamation mark indicates that changes exist with respect to the content of the respective object (e.g. section or page), a green plus indicates that a node has been added, a red cross that a node has been deleted. Within the input components, changes between both versions are visualised by a coloured header row. Added text in DOM Editor and DOM table has a green background, text which has been removed is shown with a red background.

Above the input components the respective revision is displayed with the version number, date, time, last modifier and the comment assigned to the revision ("Revision information"). The Next and Back icons can be used to switch to the previous or next revision ("Revision selection") or the previous or next change ("Show changes") to the respective object.

3.1.5.8   Special characters

Special characters can be used in a text with the help of this menu function. A special characters table opens when the function is called up.
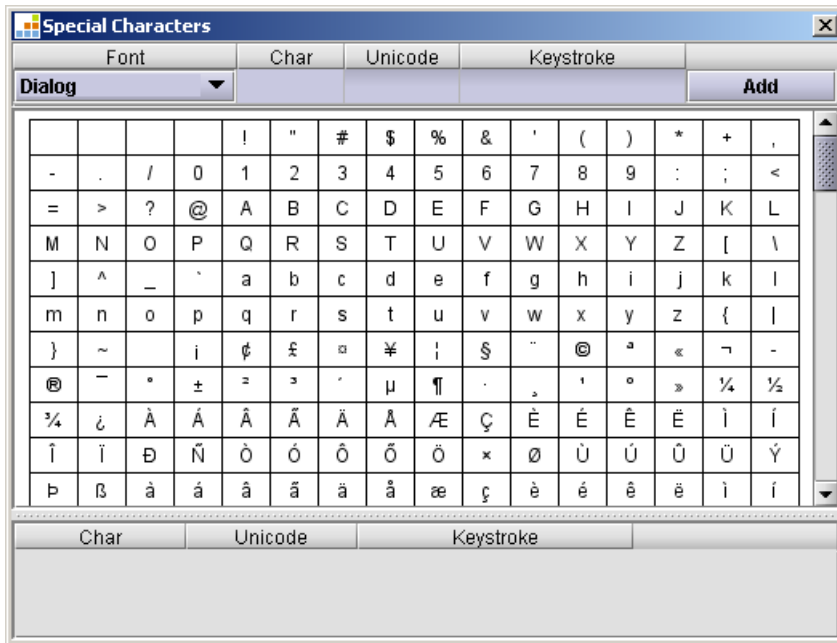
Figure 3-27: Inserting special characters

Use the "Add" button to define keyboard shortcuts for frequently used special characters. If they are no longer needed, the assignment can be undone below the special characters table using the <Del> or <Ctrl> + <Del> keys.

3.1.5.9   Media Import Wizard

When this menu function is executed a window opens with the Media Import Wizard which instructs the user step by step on how to import large quantities of media for the various languages and resolution of a project.
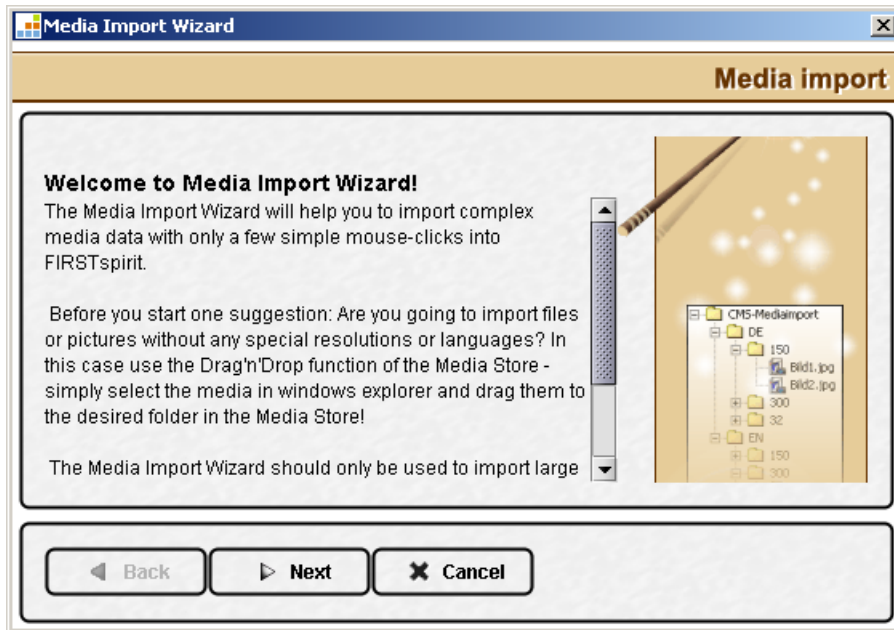
Figure 3-28: Media Import Wizard

It is possible to simultaneously import all the files collated in a folder. It is necessary to explain to the system how the media and files in the computer's file system are available for import.

Detailed documentation on use of the Media Import Wizard is given in Chapter 6.8 page 202.

> ! *If language-independent media only is to be imported with original resolution, it is possible to move the folder with the relevant media directly from the computer's file system into the Media Store using Drag & Drop. The files in the folder are then automatically imported into the Media Store folder. A differentiation is made between pictures and files and they are created accordingly. For information about the effects of media restrictions see Chapter 6.2.1 page 162.*

3.1.5.10 Send message

When this function is executed a window opens with a tabular list of all users working on the server. Each user can use this function to send a message (not a task) to other users. Double-click a user to open a further window in which the specific message can be typed.

Figure 3-29: Send message

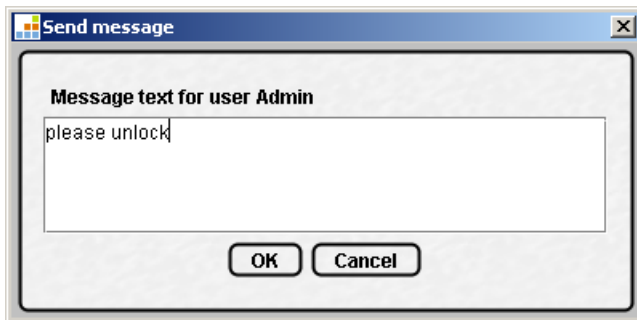The message is sent to the selected user by clicking the [OK] button and is displayed directly on their screen.

Figure 3-30: Message received

The recipient can directly reply to the message using the [Reply] button.

[Reply to all] If the message was sent to several users simultaneously, this button can be used to send a message to all recipients (except yourself) of the original message.

> ❗ *The following must be noted when using "Reply to all": Server administrators are given a Selection including all recipients, non-server administrators only the recipients logged in to the same project. The recipients selection can be limited before sending the message.*

3.1.5.11 Discussion forum

If this menu function is executed, a window opens in which all users logged onto the server can exchange messages centrally.
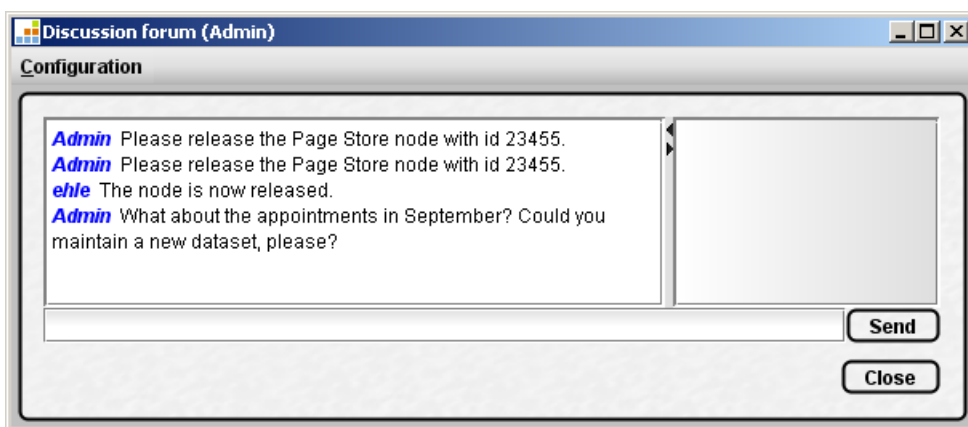

Figure 3-31: Discussion forum

3.1.5.12 Project conversion

This menu function enables project developers to automatically adjust the version differences after converting the project into Version 4.0.
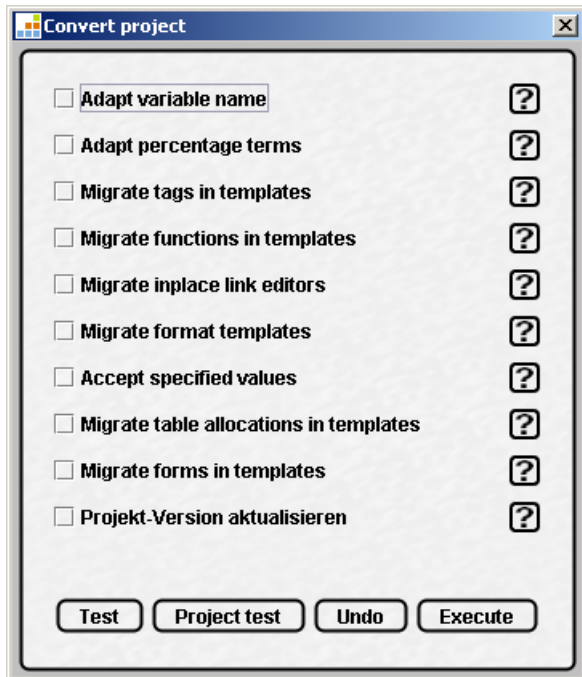
Figure 3-32: Convert project

3.1.5.13 Copy FirstSpirit address

Execute this menu function to copy the address of the activated object in the FirstSpirit client into the (temporary) clipboard.

3.1.5.14 Switch to FirstSpirit address

If this menu function is executed a window opens in which the URL address of an object in the FirstSpirit client can be entered. This makes it possible to switch directly to this object.

3.1.5.15 Execute Script

All scripts for which the user has permission to execute are listed under this menu item. Scripts enable pre-programmed actions or calculations to be executed.

3.1.5.16 Extended logging

If this menu function is activated, extended data is written in the Client Log. In addition, the activated tree element is highlighted by a yellow frame.

3.1.5.17 Print (from V4.1)

> ❗ *Using this menu item the printing function can be started from FirstSpirit Version 4.1. For further information see Chapter 3.2.11 page 103.*

3.2 FirstSpirit Toolbar

Figure 3-33: Toolbar of the FirstSpirit JavaClient

The Toolbar contains the most important functions required by an editor in their daily work with the FirstSpirit JavaClient.

> ❗ *Several functions are available, depending on the user's permissions and the position in the tree structure. These functions are deactivated in the toolbar and are shown in grey.*

> ❗ *The toolbar contains other icons to some extent. These are also described in the following.*

3.2.1 Position Forward / Position Back

Figure 3-34: One position forwards or back

The arrow buttons can be used to go back to objects in the tree structure of the FirstSpirit JavaClient which have already been selected once during the current session. The arrow on the right next to the **Back** or **Forwards** symbols enables the user to simultaneously move several steps backward or forwards. When an arrow is activated a selection of all previous objects which can be switched to again appears.

3.2.2 Update

Figure 3-35: Update

This function is used to refresh the respective store. This is necessary if several people work on a project simultaneously and make changes in the same store. If you have finished editing a task it is advisable to refresh the store before editing another object so that you are certain of having the up to date version. Otherwise it is possible that the structure shown on the left in the tree for the relevant sub-tree has changed in the meantime (e.g. new objects may have been added or previously available objects deleted) which would result in version conflicts. However, FirstSpirit usually intercepts possible version conflicts.

> *If you have edited an object and not saved it back to the server you may not use this function! Otherwise the unsaved changes would be overwritten by the server version and would therefore be lost.*

3.2.3 New Bookmark

Figure 3-36: Save and open bookmark

With this function it is possible to save frequently used objects and to open them using the selection box next to this symbol. Bookmarks always relate to a specific project and a specific user; several users of a project must each specify their "own" bookmarks.

To create a bookmark, the required object is marked in the tree structure. Click the bookmark symbol to open the following window:
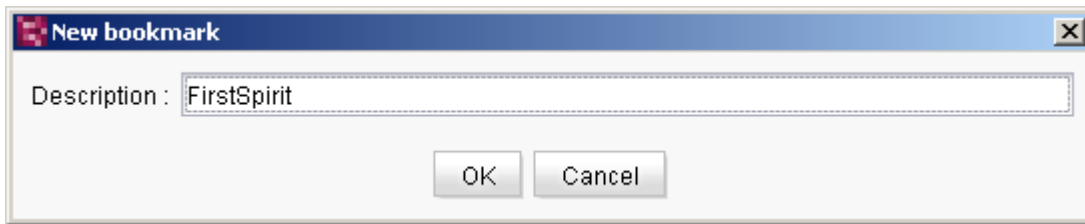
Figure 3-37: New Bookmark

The "Description" field is filled automatically and can be changed if necessary. Click OK to close the window. The new bookmark is now saved with the selected description text.

From FirstSpirit Version 4.1, only one bookmark can be created for each object. The following message appears if a bookmark is to be added to an object for which a bookmark already exists:
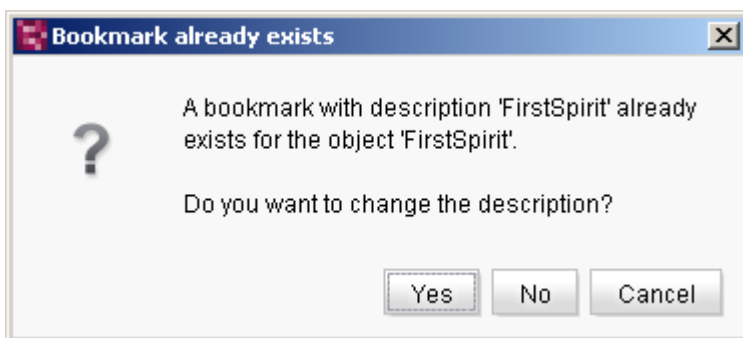
Figure 3-38: Bookmark already exists ("LightGray" Look & Feel)

Click Yes to adopt the new description for the bookmark, click No and the old description is retained.

Bookmarks can be created on any level of the respective project's tree structure. In the Content Store, bookmarks can even be set on individual data sets; however, not on data sources (content).

> ❗ *From FirstSpirit Version 4.1, this function can only be used to set a bookmark on data sources. A bookmark can be set on individual data sets using the "New Bookmark" function in the respective context menu (see Chapter 5.2.6 page 141).*

The Edit Bookmarks function can also be opened using the arrow next to the Bookmark icon.

3.2.4 Home (from V4.1)

Figure 3-39: Home

Using this button the user can jump to his personalised project homepage from each position of the actual project. It provides a personalised overview about the bookmarks, open tasks and possible actions with icon and in the colour of the respective store. For further information see Chapter 1.3.1 page 24.

3.2.5 New

Figure 3-40: Create new object

This button can be used to create new objects. The options available are always dependent on the store or level the user is currently in. E.g. if you are at menu level in the Site Store you can use this button to create a new page reference. This function is also accessible via the respective context menus and is described in greater detail in the relevant places.

3.2.6 Lock /unlock (edit mode on / off)

Figure 3-41: Block editing of an object

To make changes to an object, it is first necessary to switch on edit mode (i.e. lock the object). This prevents simultaneous editing by another user and therefore also prevents version conflicts.

After the required changes have been made edit mode must be switched back off again (i.e. the object unlocked) to release the relevant object for editing by other users.

> ❗ *Use the keyboard shortcut CTRL + SHIFT + E to quit Edit mode without adopting any of the changes made.*

Depending on the settings in the server and project configuration, a comment is asked for on exiting Edit mode (see *FirstSpirit Manual for Administrators).*

> ❗ *This function is released for FirstSpirit Version 4.1 and higher only. Screenshots are therefore displayed in the new "LightGray" Look & Feel. The display can differ slightly in the "Classic" Look & Feel.*

This function enables the editor to explain changes they have made.

The following options are possible:

Forced comment: The comment line shown in Figure 3-42 cannot be closed using OK until the editor has entered a text. I.e. Edit mode is not ended until a comment has been entered:

Figure 3-42: Forced version comment

Optional comment: The comment line can also be closed without making an entry by clicking OK.

Figure 3-43: Optional versions comment

No comment: The editor can quit Edit mode without having the comment line displayed.

Whether change comments have to be added or not can vary from store to store.

Comments entered on changing an object are also included in the object's version history (in Figure 3-44 "This is a comment!"):
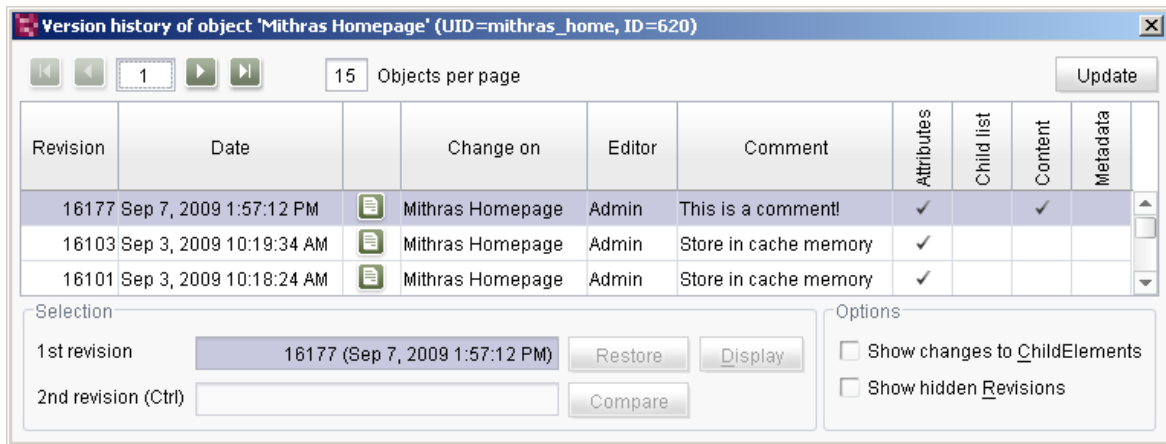

Figure 3-44: Change comment in the version history

There is no prompt for comment input if Edit mode is quit using CTRL + SHIFT + E.

When edit mode is exited all the changes made are automatically saved.

3.2.7 Save

Figure 3-45: Save

This function is used to save changes to the object currently being edited. It is important to note that this is a kind of "temporary storage". An object is normally saved automatically as soon as edit mode is exited. However, if the changes made are extensive or time-consuming it is advisable to briefly save the work done as you go along to prevent data losses as there is no automatic rapid saving in the background in FirstSpirit!

3.2.8 Preview

Figure 3-46: Preview

If, while working on an object, you want to check what the finished website will look like you can do this with the help of the Preview function. Click this button to reference the content of the page with the layout and to create a test version of the website. The page is then displayed in the selected browser. When you view a preview via the relevant context menus you can freely select from all languages assigned to the project.

It is also possible to continue working in the JavaClient while the preview is generated. Several points must be noted when requesting a preview:

1. If a browser has not been entered in the user settings in the Global Store (see Chapter 9.3.1 page 247), the preview is displayed using the default browser entered in the operating system.
2. For a preview in the Page Store the navigation is rendered according to the context of the first page reference found in the Site Store. If there is no page reference for the page the context of the root node of the Page Store is used for the rendering.
3. Press the "Ctrl", "Alt" or "Ctrl+ Alt" keys too to select entered browsers from the Global Store.
4. The navigation is integrated in the browser preview but only works to the extent that the target pages of the references have already been generated. In addition, generation of the page is highly dependent on the scope of the whole project.
5. If a re-rendering of the preview of a Page Store page is to be generated without previous deactivation of the edit mode, this can be forced using **SHIFT + Preview**.
6. The system is unable to detect changes to the Site Store which e.g. concern the navigation graphics. Additional pressing of the SHIFT key forces regeneration of the page.
7. The caching mechanisms of the browser must be switched off to make all changes visible!
8. If a "fatal" error occurs during the rendering of a page preview which results in the relevant file not being able to be generated a text file with the errors is displayed instead.
9. Special case: Preview calculation of a page with content (data source). All data records from the table are displayed in a preview in Page Store. The data record constraints set in the Site Store is only activated for previews from the Site Store.

> *If the Integrated preview is activated in FirstSpirit Version 4.2 or higher (see Chapter 3.1.4.4 page 72), then the preview can be requested in an external web browser by right-clicking the Preview icon. Likewise, right-click to display the external preview of a picture from the Media Store.*

3.2.9 Delete

Figure 3-47: Delete

Click the icon (or press the "Del" key) to delete the currently selected object or the currently selected sub-tree in the FirstSpirit JavaClient (see Chapter 3.2.9.1 and 3.2.9.4). The function can be used in the following stores:

- in the Page Store (at the level of a folder, a page or a section)

- in the Site Store (at a menu level or a page reference)

- in the Media Store (at the level of a folder or a medium)

- in the Content Store (at the level of a data table)

> *A workflow, e.g. for deleting an object, can be tied to this function from FirstSpirit version 4.1 (see Chapter 3.2.9.7 page 101). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

The arrow next to the delete icon is used to display a list of all objects deleted during the current session. The deleted object or deleted sub-tree can be reinstated by selecting an object (see Chapter 3.2.9.3 and 3.2.9.6).

> *The "Delete" and "Restore" functions can only be used if the editor has the necessary permissions ("Delete" and "Create") to the respective objects.*

Administrators have extended functions for deleting objects and sub-trees (see Chapter 3.2.9.2 and 3.2.9.5).

3.2.9.1   Delete objects

To delete individual objects the object, for example a section, is selected (marked) in the tree view. The selected object can then be deleted from the project by clicking the "Delete" icon (or using the "Del" key).

Before the deletion a confirmation prompt appears:

Cancel   The delete action is interrupted with a click. The selected element is retained in the project and the dialog is closed.

OK   Deletion of the selected object is confirmed with a click. The object is removed from the project. The user can restore the deleted object during the current session by using the "Restore" function (see Chapter 3.2.9.3).

If the object to be deleted is still being used in the project the user cannot delete the selected object. The existing uses of the object are displayed:
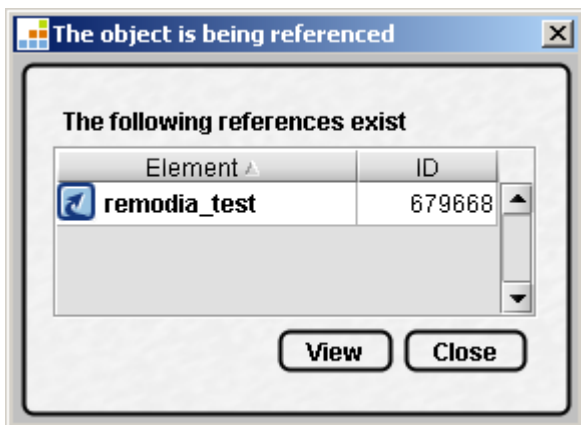
Figure 3-48: Display of the references for an object

View   Deletion of objects still in use can result in "broken links" within the project. Therefore, before an object can be deleted all its uses must be deleted. Within the table the user can select a use and click the button to directly switch to the selected use and delete it if necessary.

Close   The dialog is closed. The selected element and all uses remain in the project.

3.2.9.2   Advanced functions for administrators

If the editor has administration permissions within the project they can also delete objects which are still being used. In this case the following dialog is displayed:
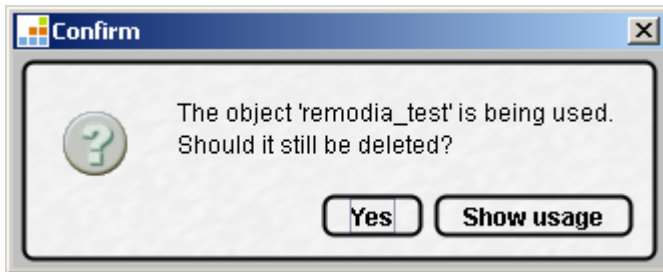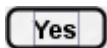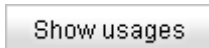

Figure 3-49: Delete object despite uses

Yes The object is removed from the project with a click, although it is still being used within the project.

Show usages Click the button to display all uses of the object selected for deletion.

3.2.9.3   Restore objects

Using the "Delete" icon: During the current session the user can restore a deleted object using the "Restore" function which is executed using the arrow key of the "Delete" icon.
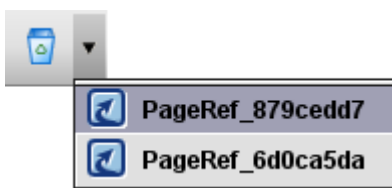
Figure 3-50: Restore deleted elements using the "Delete" icon

The function can only be used during the current session. If the user has exited the current session the delete action cannot be undone using the "Restore" function.

Using the Context Menu: However, it is possible to restore deleted objects within the individual stores at any time using the Context Menu. To do this the user must select the parent node of the deleted element in the tree view and then open the "Restore Deleted Objects" function in the Context Menu (see Chapter 4.3.3 page

122).

3.2.9.4   Delete sub-trees

Apart from simple deletion of objects, FirstSpirit also supports the deletion of sub-trees. To delete a sub-tree the object, for example a folder of the Page Store, is selected in the tree view. The selected object and all lower level objects can then be deleted from the project by clicking the "Delete" icon (or using the "Del" key).

A confirmation prompt appears analogous to the deletion of individual objects:

Cancel The delete action is interrupted with a click. The selected sub-tree remains in the project and the dialog is closed.

OK Deletion of the selected sub-tree is confirmed with a click. All elements of the sub-tree no longer used in the project are deleted from the project. The user can restore the deleted objects during the current session by using the "Restore" function (see Chapter 3.2.9.6).

If elements selected for deletion within the sub-tree are still being used in the project they cannot be deleted (cf. Chapter 3.2.9.1). The following dialog shows successful and unsuccessful delete actions on the server:
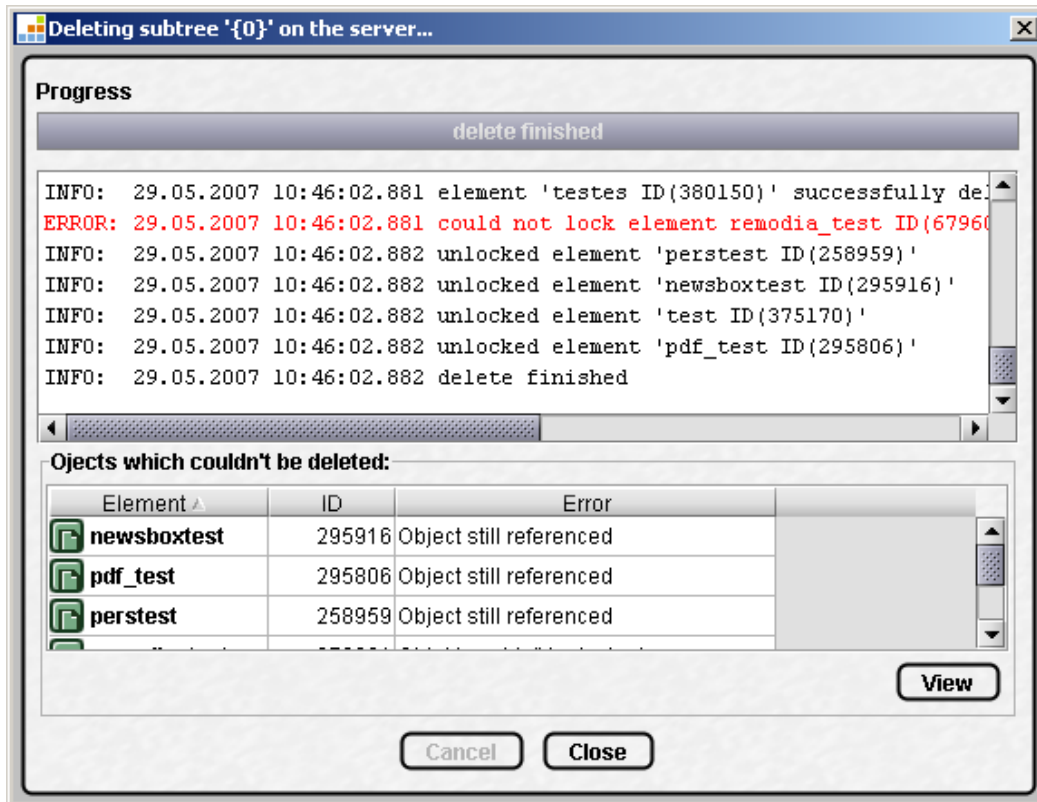
Figure 3-51: Delete a sub-tree

The status of the executed delete actions is displayed in the top part of the window. Elements which are no longer referenced are deleted from the server. Elements still being used cannot be deleted; therefore in this case an error is displayed (highlighted in red).

If a sub-tree is deleted in which at least one object is still in Edit Mode, only the part of the content not being edited is deleted and the edited objects and the folder itself are kept.

All elements of the sub-tree which could not be deleted are listed in the bottom part of the window. These elements and their parent nodes remain in the project.

View If an undeleted object is selected in the table, click the button to switch directly to this object.

3.2.9.5 Advanced functions for administrators

If the editor has administration permissions within the project they can also delete objects which are still being used. In this case the following dialog is displayed:

Figure 3-52: Delete a sub-tree (for administrators only)

`Delete` All objects of the sub-tree are removed from the project with a click, irrespective of whether they are still being used within the project or not. This can cause "broken links".

`Skip` This function is implemented analogous to the deletion of objects without administration permissions. Only the elements of the sub-tree no longer being used in the project are deleted.

`Cancel` The delete action is interrupted with a click. The selected sub-tree remains in the project and the dialog is closed.

3.2.9.6   Restore sub-trees during a session

Using the "Delete" icon: It is not possible to use the "Restore" function to restore a sub-tree or individual elements of a sub-tree during the current session (cf. Chapter 3.2.9.3).

Using the Context Menu: However, it is possible to restore deleted objects within the individual stores at any time using the Context Menu. To do this the user must select the parent node of the deleted element in the tree view (at the highest level, i.e. the root node) and then open the "Restore Deleted Objects" function in the Context Menu (see Chapter 4.3.3 page 122).

3.2.9.7   Connecting a workflow to the delete function (from V4.1)

> **!** *This functionality is released only from FirstSpirit version 4.1 on.*

A project-specific workflow can be created and directly connected with the controls provided to date for deleting elements (menu bar buttons, context menu entry) to

delete elements in the FirstSpirit JavaClient and in the FirstSpirit WebClient. Instead of simply deleting an object, for example a page, a more complex delete function can be provided via the workflow, for example the additional deletion of dependent objects of a page.

The configuration is defined in the project developer, the workflow is then started within the client using the familiar control elements. The following controls are available for this:

- Select element and click <Del> key.
- Select element and click the "Delete" context menu entry
- Select element and click the ![icon] icon in the icon bar

Analogous to the multiple selection of workflows, it is possible to delete a set of objects at the same time via a workflow (see Chapter 12.2 page 423).
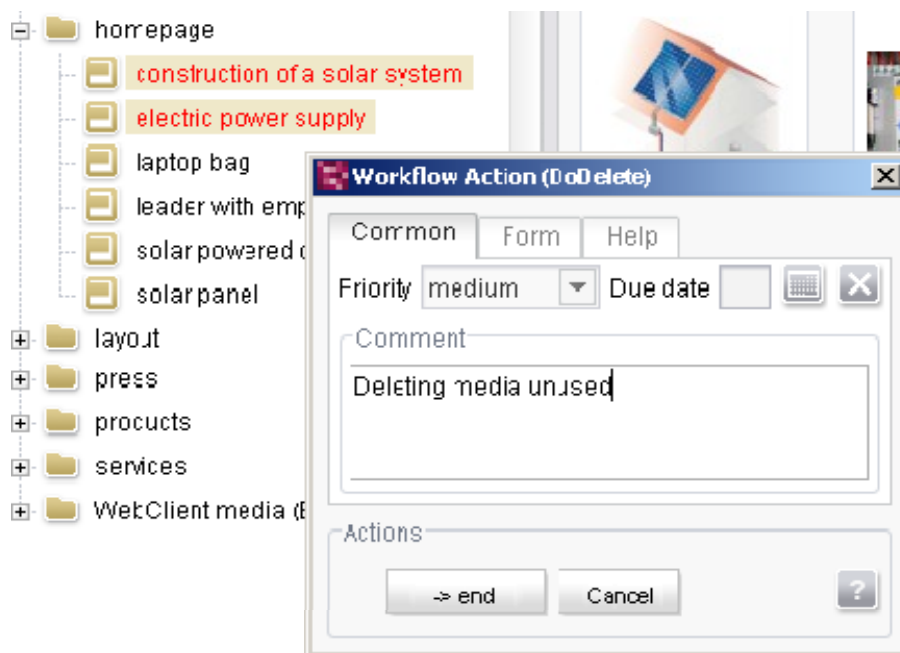
Figure 3-53: Deleting several objects by means of a workflow

For further information about the deleting workflow see *FirstSpirit Manual for Developer (Basics)*.

3.2.10   Show/Hide Task List

Figure 3-54: Show or hide Task List

This function can be used to show or hide the Task List. (The Task List is described in detail in Chapter 11.1 page 314!)

3.2.11   Print

Figure 3-55: Print

> ! *The printing function can be started from FirstSpirit version 4.1 only via the menu **Extras** (see Chapter 3.1.5.17 page 88).*

This function can be used to print out parts of the tree structure in the FirstSpirit JavaClient. Everything from the activated node is displayed in the print view, depending on the current position in the tree structure.

This function is only active in the stores for content (Page Store), structure (Site Store) and global settings (Global Store).

3.2.11.1   Print preview window
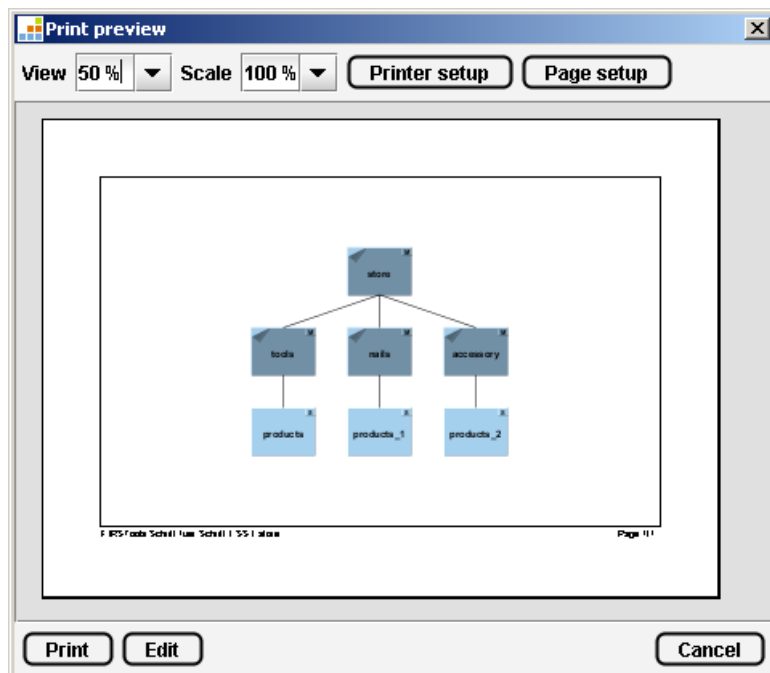
Figure 3-56: Print preview

View: Here the user can set the size of the pages to be displayed in this window. Several pages can be displayed simultaneously if the chosen display is small enough.

Scaling: The size of the displayed tree elements in the printout can be selected in this field.

Printer setup Click this button to open a window in which several print settings can be entered.

Page setup Click this button to open a window in which several settings can be entered for the printed pages.

Print Click this button to start printing

Cancel Click this button to cancel printing

Edit Click this button to open a further window in which the selected sub-tree can be limited or specific elements can be highlighted:

3.2.11.2 Edit Print Preview window
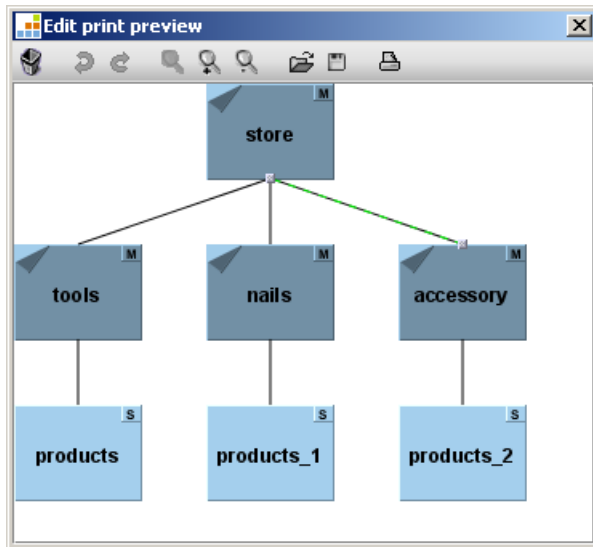
Figure 3-57: Edit Print Preview

 Delete selected elements; this icon is used to remove all the selected tree elements from the Print Preview.

 Undo; this icon can be used to undo a range of changes made in the Print Preview.

 Restore; use this icon to restore changes which have been undone.

 Default size 1:1; use this icon to display all elements in the Edit window in their original size again.

 Enlarge; this icon can be used to enlarge the display of the elements in the Edit window.

 Reduce; this icon can be used to reduce the size of the display of the elements in the Edit window.

 Load preview from the hard disk; use this icon to load a print preview stored in the computer's file system and edit it if necessary.

 Save preview on the hard disk; this icon can be used to save the current print preview in the computer's file system.

 Print preview; use this icon to return to the Print Preview window.

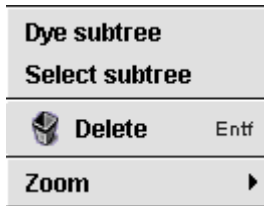### 3.2.11.3 Context menu in the Edit Print Preview window

Figure 3-58: Context menu in the "Edit Print Preview" window

Colour sub-tree: After activating this function a colour scheme appears; all selected elements are marked with the colour selected here.

Select sub-tree: This function can be used to select a tree element including all the elements below it to e.g. move this whole sub-tree to another position in the Print Preview.

Delete: Activate this function to remove all the selected tree elements from the Print Preview.

Zoom: This function can be used to change the display size of the elements.

3.2.12 Online Help

Figure 3-59: Opening Online Help

Activating this button opens a PDF file of this documentation. The relevant chapter in the documentation is displayed depending on which store you are currently in.

3.3 Search dialog (V4.2 and higher)

The search dialog for quick text search has been considerably enhanced for FirstSpirit Version 4.2 and can be opened using the input field in the top left-hand area of the JavaClient:
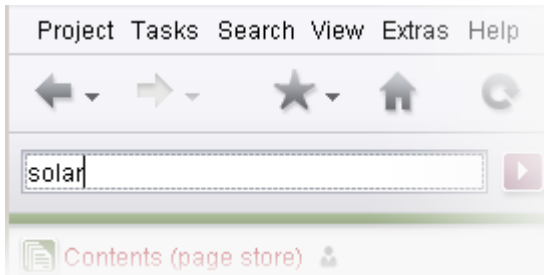
Figure 3-60: New search dialog for the quick text search

The search results are displayed in the "Search" tab after having entered a seach term and started the search:
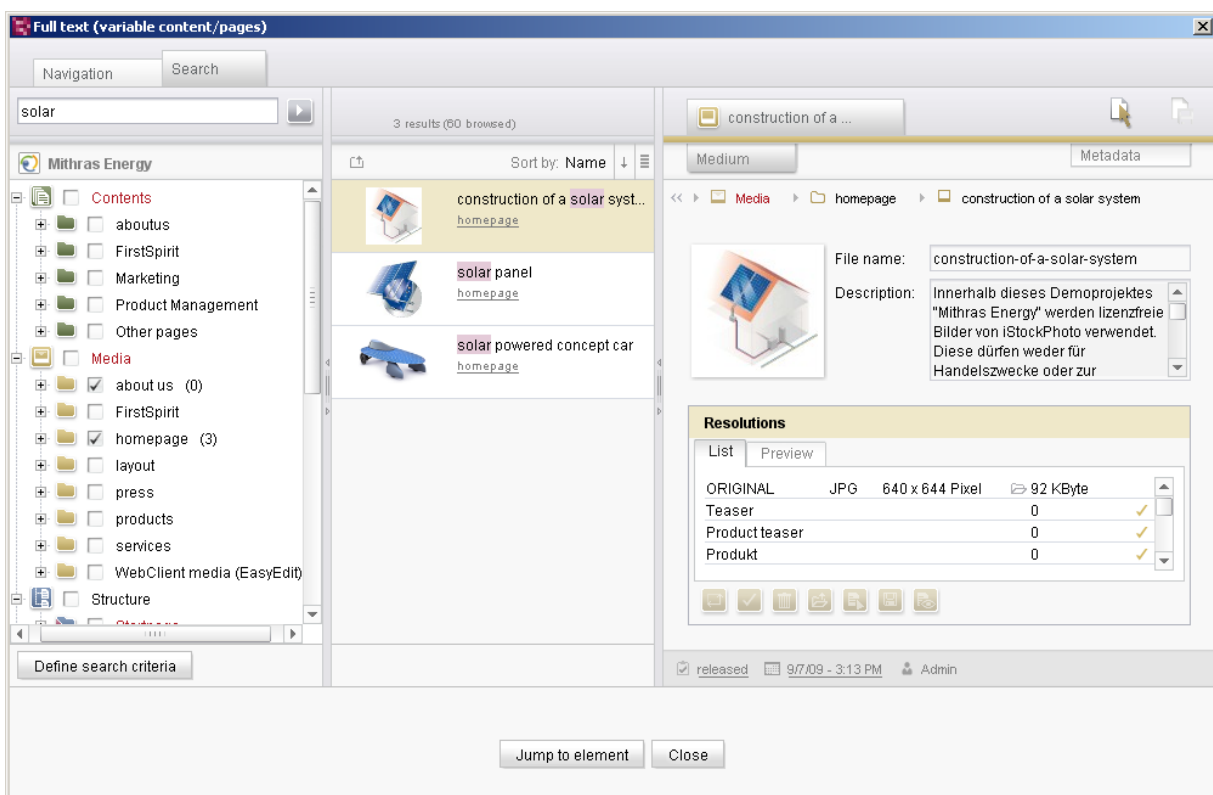

Figure 3-61: Presentation of the search results

The left-hand area shows the Stores and the individual folders with the number of search results found, e.g. (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.

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.

These functions are already familiar from the media selection dialogs introduced with FirstSpirit Version 4.1 (see Chapter 11.8 page 360).

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 11.4 page 318).

3.4 Central collection of bugs and system report (V4.2 and higher)

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. After closing the dialog or when marking exceptions as "seen" they will be moved to the tab Register "Exceptional events history". The number of exceptions is shown in brackets on the respective tab.
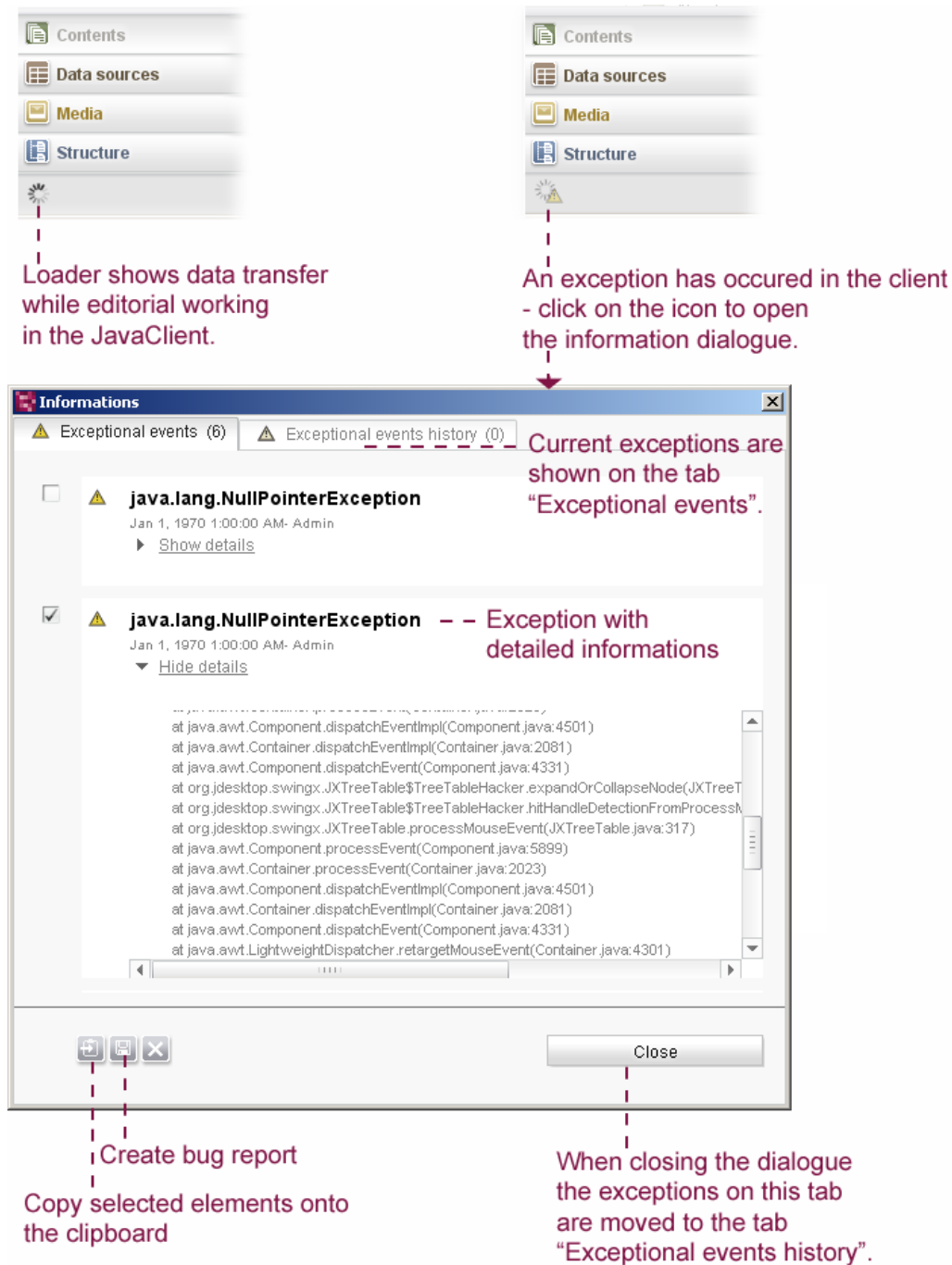
Figure 3-62: Central collection of bugs in the FirstSpirit JavaClient

One or more error messages can be selected in the information dialog and copied to the clipboard and/or subsumed in a bug report.

Copy selected elements onto the clipboard: this icon is used to copy the error messages of the selected elements onto the clipboard; they can then be subsequently further edited in other text programs.

🖫 Create report: this icon can be used to generate a system report in HTML format on the tab's selected elements. Apart from the information on the exception, such a report also contains other information which can be relevant for reproduction of the error:

- Project Settings,
- Operating system and server configuration,
- JavaClient configuration (for example, settings for integrated preview and for the browser engine used),
- Object (technical and editorial information, cf. Chapter 4.4.10 page 128)

First Spirit System-Report

Sep 28, 2009 2:23:47 PM
User: (Admin)

Comment:

Comment

Digest

System-Informations
    Common
    Project & User
    Client-Configuration
Object-Informations
    Editorial
    Technical
Exceptional events

System-Informations

Common

Client-Version	4.2_DEV.102.34113
Licensee	e-spirit
Server	batida:4300 (HTTP)
Server Version	4.2_DEV.102.34113
Java version	1.6.0_05 32bit Sun Microsystems Inc.
Operating system	Windows XP 5.1 x86
Memory used	86.03 Mbyte
Memory allocated	127.06 Mbyte
Project loading time	84.55 s
Connection speed	2.77 kbyte/s

Figure 3-63: Generated system report (extract)

☑ Select All: this icon can be used to select all the elements of the tab.

☐ Deselect All: this icon can be used to cancel selection of all the tab's elements.

[X] Mark selected elements as seen: this icon is used to move the selected elements onto the "Error History" tab.

[trash icon] Remove selected entries: this icon is used to remove the selected elements from the information dialog. They are no longer displayed the next time the tab is opened.

Close: if an exception has not been removed using the [trash icon] icon, it is retained within the current user session on closing the error dialog using the corresponding button and is displayed in the "Error History" tab the next time the info dialog is opened.

From V4.2R4 a new charging icon is used, eg in integrated preview:
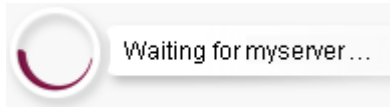
In addition, the preview is displayed faded.

This icon is also used in the bottom left-hand corner, in which error messages that occur are also displayed. If an error message occurs, it is now displayed in a kind of speech balloon, the load icon contains a number, which represents the number of error messages that exist:
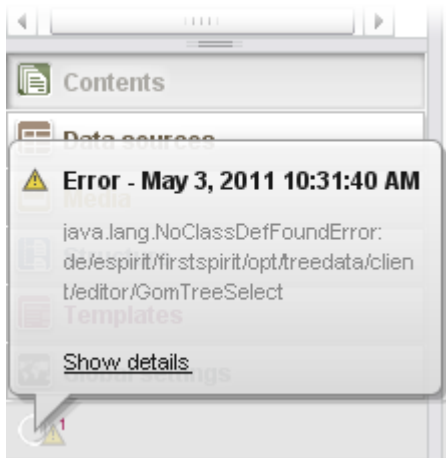

Abbildung 3-64: Error message in 4.2R4

Click "show details" to open the dialog with the complete error message/s as usual.

4   JavaClient Page Store

The Page Store is used to maintain editorial content. Here the editor creates pages and sections with editorial content which are later referenced to the structure and layout from the Site and Template Stores. The content is inserted in the pages or sections using standard input elements, for example a text box or a picture input component. The layout is individually specified for each project using page or section templates and is not the task of the editor.

The following elements can be created within the Page Store:

- Folders for structuring the pages. In most cases it is useful to copy the folder structure from the Site Store.
- Pages
- Sections / Content
- Section references

4.1   "Drag & Drop" in the Page Store

> ⚠️   *The drag & drop functions up to FirstSpirit version 4.1 including are described in the following. For a detailed documentation of the enhanced drag & drop functions from FirstSpirit version 4.2 on see Chapter 11.4 page 318.*

4.1.1   Move using "Drag & Drop"

Folders, pages and sections can be moved within the Page Store with the help of the mouse using "Drag & Drop" (indicated by a small rectangle on the mouse pointer).

When moving sections, reference names of sections and section references must always be unique within a page. If moving sections would result in double reference names, the reference name will be automatically made unique.

Figure 4-1: Example – Moving a section with the same reference name

> [!] *If the function "Confirm move operations" in the menu "Extras" is active, for each move a security dialog box must be confirmed (see Chapter 3.1.5.2, page 78). This does not apply to folders and pages which are moved from the Page Store to the Site store to create new menu levels or page references.*
>
> *Sections can only be moved if no other user is currently editing the page.*

4.1.2 Copy using "Drag & Drop"

Further, it is also possible to copy pages [icon], sections [icon] and section references [icon] in the Page Store using the mouse and by pressing the Ctrl key at the same time (indicated by a small plus on the mouse pointer).

When copying sections, reference names of sections and section references must always be unique within a page. If copying sections would result in double reference names, the reference name will be automatically made unique.
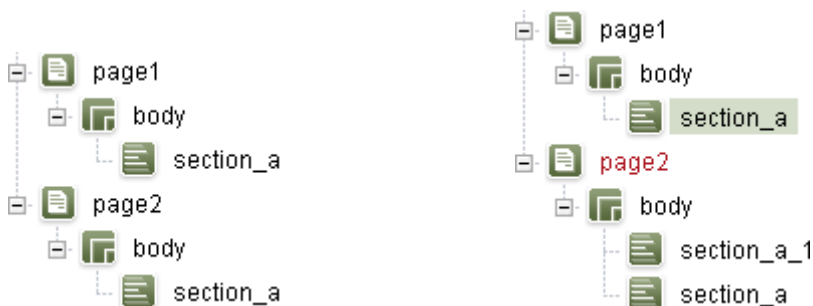

Figure 4-2: Example – Copying a section with the same reference name

> ⚠ *The user must have the necessary permissions to "Drag & Drop" (move, copy) from nodes in the Page Store. Otherwise an error message appears "This action is not possible (lack of permissions)!"*

4.1.3   Create section references

by moving the section 📑 to be referenced ("source section") using your mouse by simultaneously pressing the SHIFT and Ctrl key (denoted by a small arrow on the mouse).

> ⚠ *Section references can not be created for content (data sources).*

If **metadata** exist in the source section, these will be assumed for the section reference in FirstSpirit Version 4.2.213 and higher. No metadata can be defined. Before version 4.2.213 metadata was not assumed or – if available – fallback values (siehe Chapter 11.10 page 386), which are defined by the project developer for the respective input component. For more information about the evaluation of metadata see Chapter 11.3 page 317.

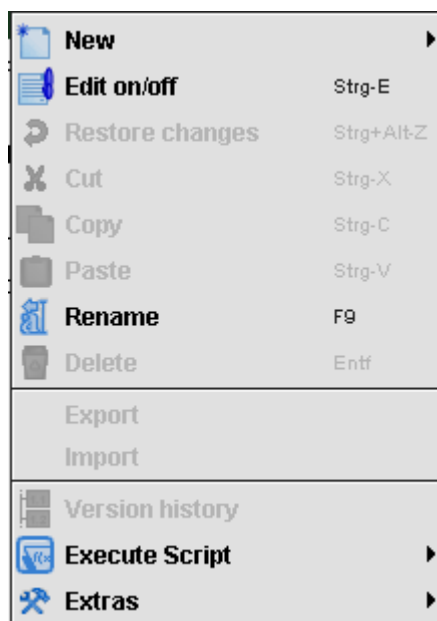## 4.2   General context menus in the Page Store

Figure 4-3: Context menu – General Page Store functions

The Page Store context menus are described in the following chapters:

1. All context menus are structured in the same way:

- the top part contains general functions

- these are followed by specific functions for the selected nodes

- The bottom area contains management functions which are usually only required by project administrators. Most of these cannot be executed by normal users and are therefore displayed in "grey".

2. To open a context menu an object, for example a folder or a page, is selected in the tree view on the left-hand half of the screen and is then right-clicked (i.e. right-hand mouse key is pressed) or using the the Application key ▤ (if available) to open the context menu for this node. The required menu item can be selected by clicking the left-hand mouse key (left-click).

3. Deactivated menu items are "grey". In this case the function is not available to the user. Possible reasons for this are:

- the object is currently being edited by another editor

- the status of the current object

- The user does not have the necessary permissions to execute a specific action.

4.2.1 New

The "New" context menu entry can be used to insert new objects into the project. The selection available depends on the object type on which the context menu was opened:

4.2.1.1 New: Root node and Folders

Figure 4-4: Selection at root node or folder level

New – Create Page This function is used to create a new page in the current position in the Page Store. After clicking a window opens in which all the available page templates are displayed.
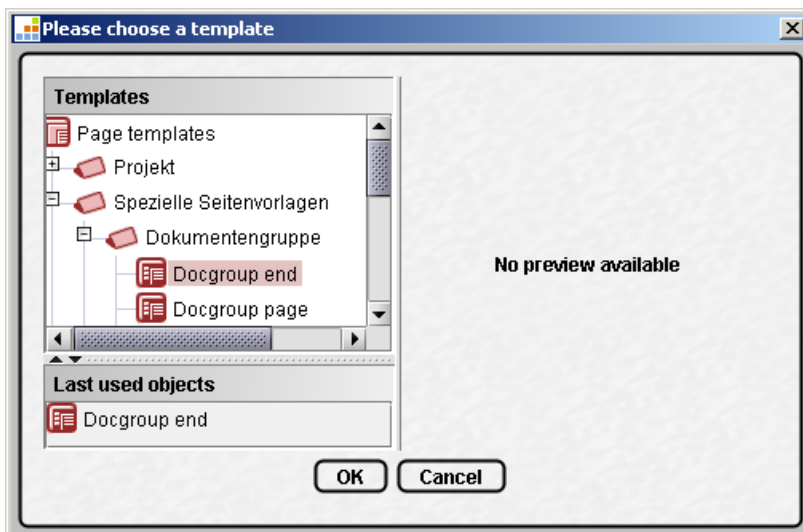

Figure 4-5: New – Select Page

You only have to select the required page template from the tree structure and confirm your selection with OK. Below the tree structure there is a list of the most recently used page templates of the respective user; this is intended to make it

easier to find frequently used templates. It is then only necessary to assign a name for the new page.

New – Create Folder: To improve clarity, the pages should not all be listed under each other but filed in folders with content that belongs together. These folders can be hierarchically arranged as e.g. in Windows Explorer. This function can be used to insert a new folder in the tree structure. Click to open a dialog box in which you can enter the folder name.
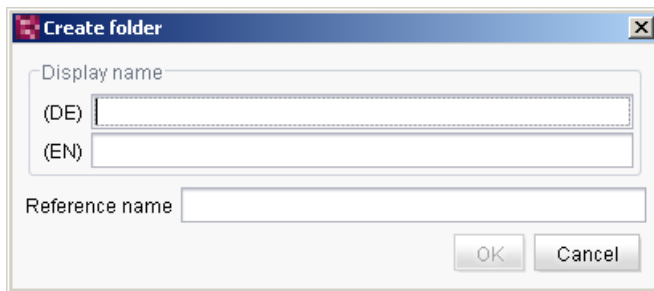

Figure 4-6: New – Create Folder

A **language-dependent display name** can be assigned to the new folder, for each editing language defined by the project administrator. Either the display names or the reference name are then displayed in the tree view, depending on the setting in the "Extras – Preferred Display Language" menu, from FirstSpirit Version 4.2 the "View – Preferred Display Language" menu (see Chapter 3.1.4.2 page 69). The Reference Name field is automatically filled with the value entered for the first display name by the editor, but can be changed (up to the initial creation of the object). The reference name may not contain any spaces, special characters or symbols. This is taken into account accordingly when the field is filled automatically.

> ! *The Reference Name field is only displayed if this setting was configured accordingly by the project administrator.*

> ❗ *In FirstSpirit Version 4.2R4 and higher, the administrator can define rules with which special characters in reference names are automatically transformed into valid characters. The conversion takes place directly during input when creating a FirstSpirit object or changing a reference name (context menu: "Extras" / "Change reference names").*
> *Special characters for which the administrator has not defined a replacement rule cannot be entered in the "Reference name" field.*
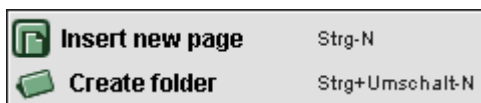
4.2.1.2   New: Page

Figure 4-7 Selection at page level

New – Create Page: This function is used to create a new page in the current position in the Page Store. After clicking a window opens in which all the available page templates are displayed. (See Figure 4-5: New – Select Page on Page 116.) You only have to select the required page template from the tree structure and confirm your selection with OK. Below the tree structure there is a list of the most recently used objects of the respective user; this is intended to make it easier to find frequently used objects. It is then necessary to assign a name for the new page.

Create New Folder: This function can be used to insert a new folder in the tree structure. Click to open a dialog box in which you can enter the folder name. (See Figure 4-6: New – Create Folder on Page 117.)

4.2.1.3   New: Content Area, Section and Content

Figure 4-8: Selection at the level of a content area of a section

New – Insert Section: This function is used to insert a new section in the content area of a page. After clicking a window opens in which all section templates allowed for the current page are displayed.
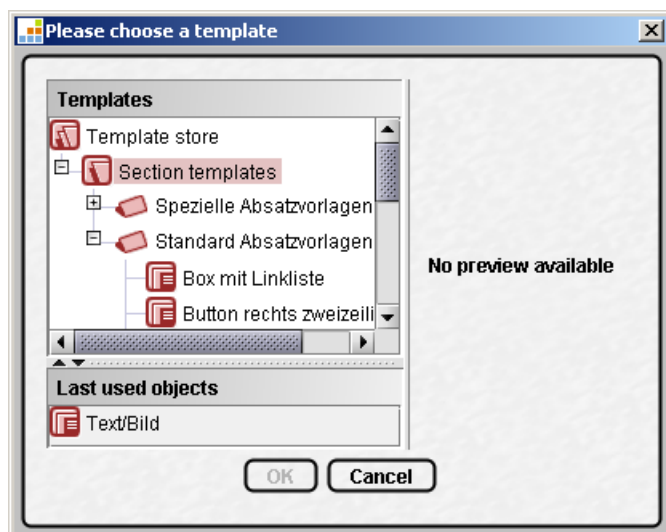
Figure 4-9: New – Select Section / Table Template

You only have to select the required section or table template from the tree structure and confirm your selection with OK. Below the tree structure there is a list of the most recently used section templates of the respective user; this is intended to make it easier to find frequently used templates.

If this function is executed in a content area and if this content area already contains other sections the new section is always automatically inserted in the first position, i.e. in front of all other sections.

If this function is executed in a section or content the new section is always inserted after the selected section.

If a new section has been inserted Edit Mode is automatically activated for the page in which the section is located. In this way content can be directly inserted.

The display name (of the default editing language) of the selected section template is automatically adopted for new sections added. The reference name is formed from the display name and is also set automatically. At the same time, any invalid characters (e.g. special characters, symbols, spaces, capital letters) are removed from the reference name. As the reference name must be unique within a page, this value is uniquely assigned by appending numbering.

4.2.2 Lock/Unlock (Edit Mode On/Off)

This function is used to activate Edit Mode for the selected node. No other editor can then make changes to this object because it is locked.

Opening this function again deactivates edit mode (unlocks the object), i.e. View Mode is reactivated.

4.2.3 Reset Changes

This function can be used to undo changes made during the current editing process and which have not yet been saved.

4.2.4 Cut

This function is used to cut the current object and store it in the (temporary) clipboard. It can be inserted again in another position in the tree structure.

4.2.5 Copy

This function is used to generate a copy of the current object and store it in the (temporary) clipboard. Copies can be generated pages and sections.

4.2.6 Insert

This function is used to insert the content of the (temporary) clipboard in the current position of the tree structure. This function is only active if there is data in the (temporary) clipboard which may be inserted in the current position.

4.2.7 Rename

This function can be used to change the name of the current object in the tree structure of the FirstSpirit JavaClient. After the function is executed a window opens with the object name to date; this can now be changed.

4.2.8 Delete

This function can be used to delete the current object from the tree structure of the FirstSpirit JavaClient. Accidental deletion is prevented by a confirmation prompt.

> ❗ *A workflow, e.g. for deleting an object, can be tied to this function from FirstSpirit version 4.1 (see Chapter 3.2.9.7 page 101). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

4.2.9 Version history

A window opens in which all the old versions of the current object are listed. For precise documentation of version history in the Page Store see Chapter 11.12.3 page 405.

4.2.10 Workflow

If workflow is not yet active for the selected object, all workflows defined in the permissions system for these nodes in the tree structure are listed under this menu item. The required workflow can be started under this menu item.

If a workflow is already active for the selected object it can be switched to another workflow action/state under this menu item.

Detailed documentation of workflows is given in Chapter 12 page 422.

4.2.11 Execute Script

All scripts which can be opened in this position in the JavaClient are listed under this menu item. Scripts enable pre-programmed actions or calculations to be executed.

4.3 Special context menus in the Page Store

Export
Import
Restore deleted objects

Figure 4-10: Context menu – Special functions at folder level

4.3.1 Export (at folder and page level)

This feature is only available to administrators.

This function can be used to export the selected folder or the selected page with all the necessary information to the hard disk, from where it can later be imported again in another project. A window opens with the file structure of the computer to enable you to specify a suitable storage location for the export.

> **!** *The "Export" function is a client-side function and therefore sets substantial requirements for the main memory of the client system when large quantities of data are involved. It should therefore only be used to export small quantities of data.*

4.3.2 Import (at root, folder and page level)

This function can be used to import an export file with all the necessary information back into the project. A window opens with the file structure of the computer enabling you to select the required export file.

> **!** *The import function on root node is available only from FirstSpirit version 4.1.*

4.3.3 Restore Deleted Objects (at root, page, content area and folder level)

The Restore Deleted Objects function can be opened both at root, page, content area and at folder level. If a folder, page or section has been accidentally deleted from the tree structure this function can be used to restore the deleted object. After clicking a window opens with the deleted objects.

Revision ▽	ID	Name	Object count	deleted on	deleted by
2016	287443	287443	1	9/29/06 3:29 PM	Admin (Admin)
2015	287442	287442	1	9/29/06 3:25 PM	Admin (Admin)
1842	287274	287274	1	8/28/06 4:07 PM	Admin (Admin)
1815	287267	287267	1	8/25/06 11:04 AM	Admin (Admin)

Figure 4-11: Deleted objects

All objects for which a backup exists are displayed at root, page and content area level, while at folder level only the objects located underneath this folder are

displayed. The following information is given for each object:

Revision: Version number of the deleted object.

ID: The unique ID number of the deleted object

Name: The reference name of the deleted object.

Number of objects: The number of objects located in the tree structure below the deleted object. These hierarchically lower level objects are also inserted again by the restore function.

Deleted on: Date and time when the object was deleted.

Deleted by: Name of the user who deleted the object.

[Restore] To restore, it is only necessary to select the required object and press the button.

Wherever possible, restored objects are inserted in the same position in which they were located before being deleted. If objects are restored at root level, a dialog can be used to select the restore position. Here too, wherever possible, the position before deleting is displayed.

However, if e.g. a folder in which a page to be restored has also been deleted (and is to remain deleted), this page is inserted directly below the tree node at which this function was opened.

Preview ▶
Show preview errors ▶
Export
Import
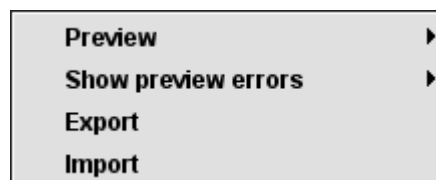
Figure 4-12: Context menu – Special functions at page level

4.3.4   Display Preview (at page and content area level)

This feature is only available to administrators.

This function can be used to check how the page just edited is displayed on the website. Opening the function generates a preview of the page. This function can be carried out separately for all editorial languages as well as for the current and the released state of the page / content area.

The button [image] in the toolbar is available for this function too! (See Chapter 3.2.8 page 94.)

4.3.5   Display Preview Errors (at page and content area level)

This function can be used to display again any errors which occurred during the last preview rendering.

Figure 4-13: Display preview errors

This function can be carried out separately for all project languages as well as for the current and for the released state.

4.3.6   Change Position (at section level)

Figure 4-14: Context menu – Special functions at section level

Each section can be moved in its position relative to the other sections to enable subsequent insertion of new sections and better handling when changing their order where larger page restructuring is involved.

The position change involves changing one position **up** or **down** or, helpful for pages with many sections, directly at the **first** or **last** position.

4.4 Administrative context menus in the Page Store

Release
Delete Metadata
Change Permissions
Reset write lock
View Template
Show usages   Strg-U
Cancel editing

Figure 4-15: Context menu – Administrative functions at page level

4.4.1 Extras – Release (at root, folder and page level)

This function can be used to directly release the current object or a specific release option can be used.

Detailed documentation on the specific release options is given in Chapter 12.4 page 433.

4.4.2 Extras – Delete Metadata (at root, folder and page level)

Execute this function to delete all metadata defined for the current level in the tree structure.

4.4.3 Extras – Change Permissions (at root, folder and page level)

This function can be used to define the permissions for the current nodes in the tree structure.

Detailed documentation on the definition of permissions is given in Chapter 13.1.2 page 445.

4.4.4 Extras – Reset Write Lock

If a write lock exists for the selected node due to an active workflow the write lock can be deleted or cancelled using this function. (The write lock is indicated by italic lettering in the tree)

4.4.5 Extras – View Template (at page and section level)

When this function is executed the template is displayed in the Template Store used for the current page or current section.

4.4.6 Extras – Show Usages (at page level)

This function can be used to determine where in the Page Store the current page is referenced. A window opens in which all page references are listed.

Click one of the references and the FirstSpirit JavaClient goes directly to the corresponding position in the Site Store.

Delete Metadata
Change Permissions
Reset write lock
View Template
View source section
Edit validity period
Show usages   Strg-U
Cancel editing
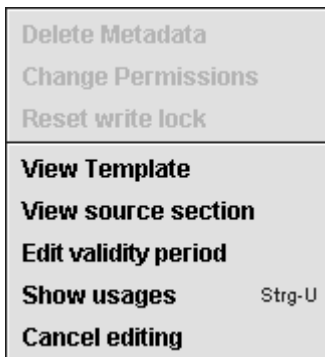
Figure 4-16: Context menu – Administrative functions at section level

4.4.7 Extras – change reference name (at root, folder and page level, V4.1 and higher)

Each FirstSpirit object has a reference name, which must be unique in each store. Each object can be identified by its reference name.

> ! *Reference names are only shown in the project if the "Reference names" setting has been activated in the "Extras / Tree Display" menu, in FirstSpirit Version 4.2 and higher: "View – Preferred display language" (see Chapter 3.1.4.2 page 69).*

In general, reference names are issued automatically when objects are created and are based on the display name. These reference names can be subsequently changed using the "Change reference names" menu item. However, the user must have "Change" permission for the respective object. The reference name should only be changed if the object is not yet referenced in the project (e.g. a picture has not yet been used on a page or in a section of the Page Store or a page of the Page Store is

not yet used in the Site Store ("page reference")), as otherwise the existing reference becomes invalid. The following dialog is displayed:

Figure 4-17: Change reference name

If the reference name is changed despite an existing reference, it may be necessary to restore the reference manually using new selection.

In FirstSpirit Version 4.2R4 and higher, depending on the project administrator's settings, the menu entry may be greyed out regardless of the "Change" permission; in this case the reference name cannot be changed.

4.4.8   Extras – Jump to source section (at section reference level)

Using this function the referenced section will be displayed in the tree structure and can be edited if required (and if authorized).

4.4.9   Extras – Edit Validity Period (at section level)

This function can be used to specify in which period a section should be presented on the website.
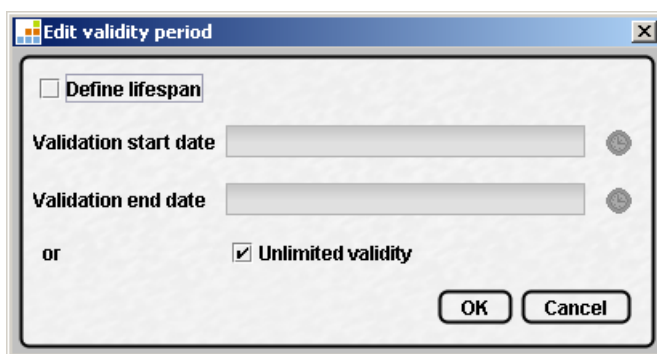
Figure 4-18: Edit validity period

Define validity period: Activate this option to enable a validity period to be defined for the selected section.

Validity Start: Click the 🕐 icon behind the line to open a new window in which the validity start can be set to the nearest minute:

Figure 4-19: Define validity period

Validity End: The end of the validity can then be set in the same way by clicking the 🕐 icon behind the line.

Unlimited validity: If this option is active, the selected section is valid for an unlimited period from the validity start. Definition of a validity end is then ignored.

Outside this period the section is not generated on the website. If a validity period has been defined it is displayed in the status bar as soon as the section is selected in the tree view.

4.4.10 Extras – Display Properties (from V4.2)

This function can be used to show technical and editorial information on individual project content in a separate dialog and to collate it into a system report. It can also be opened using the keyboard shortcut Alt + P. The information can vary depending on the object type.
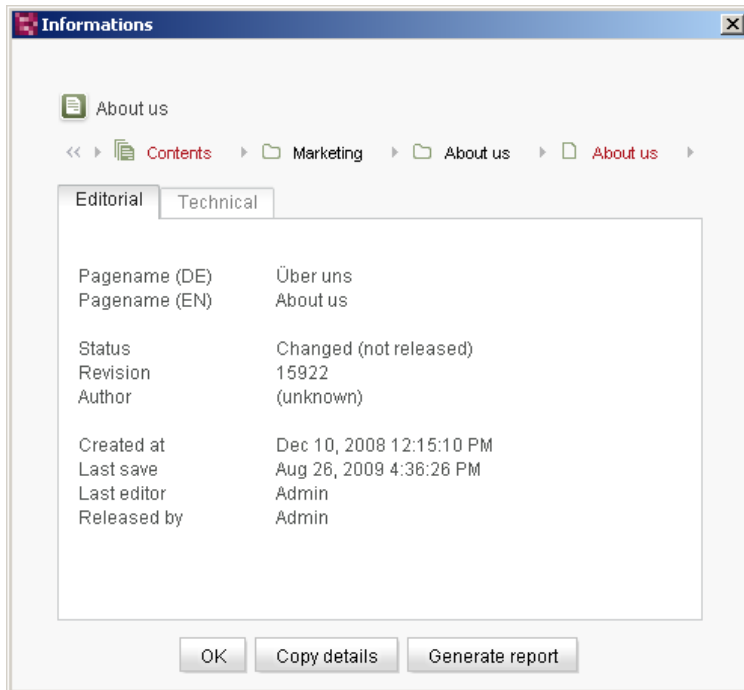
Figure 4-20: Properties of a page – editorial

The path can be used to display the properties of other objects.

Editorial tab

The editorially relevant properties of an object are displayed in this tab:

Pagename: Display name of the object (language-dependent)

Status: displays the status (e.g. "Not released", "Released", "Changed (not released)")

Revision: displays the revision

Author: Name of the user who created the object.

Created at: Date and time at which the object was created in JavaClient

Last save: Date and time at which the object was last saved

Last editor: Name of the editor who most recently edited the object

Released by: Name of the user who released the object
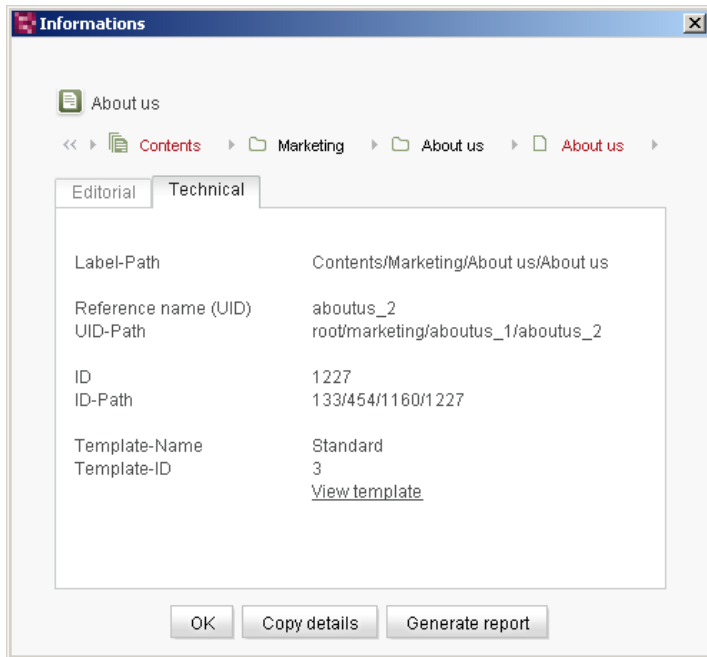
Figure 4-21: Properties of a page – technical

Technical tab

The technically relevant properties of an object are displayed in this tab:

Label-Path: Path to the current object (display names)

Reference name (UID): Reference name (UID) of the object

UID-Path: Path to the current object (reference names)

ID: ID of the object

ID-Path: Path to the current object (IDs)

> ! *The path information can also be requested using the keyboard shortcut Ctrl + Shift + Q.*

Template-Name: Display name of the template on which the object is based

Template-ID: ID of the template on which the object is based

Depending on the object type, a "Display Template" reference is output, with which it is possible to switch directly to the template on which the object is based.

Use the "OK" button to close the dialog. Use the "Copy Details" button to copy all the dialog's information onto the clipboard. Use the "Generate Report" button to output the information as an HTML page in the form of a system report (see Chapter 3.4 page 108). In addition, an additional comment can be entered, for example, an error description.

4.4.11 Extras – Cancel editing (at section level)

Using this function the editing mode can be determined, without accepting changes which have been not saved yet.

4.4.12 Extras – Display dependencies (at page and section level, from V4.1)

Using this function the reference graph can be opened for the respective object.

For a detailed documentation of the reference graph please refer to Chapter 11.13 page 413.

4.5 Settings at page level

After a new page has been inserted in the Page Store, edit mode is automatically activated for this page. The page can now be edited.

Figure 4-22: Page view

Page completely translated in this language: If the project is a multilingual project, activating this option indicates that the page is already completely available in the corresponding language. If this option is deactivated the page is not taken into account in the corresponding language when the project is generated.

In addition, there may be various input components on this level which the template developer has earmarked for this page. Detailed documentation on the input components available is given in Chapter 10 page 249.

4.6 Settings at section level

After a new section has been inserted in the content area of a page, edit mode is automatically activated for this page. The section can now be edited.
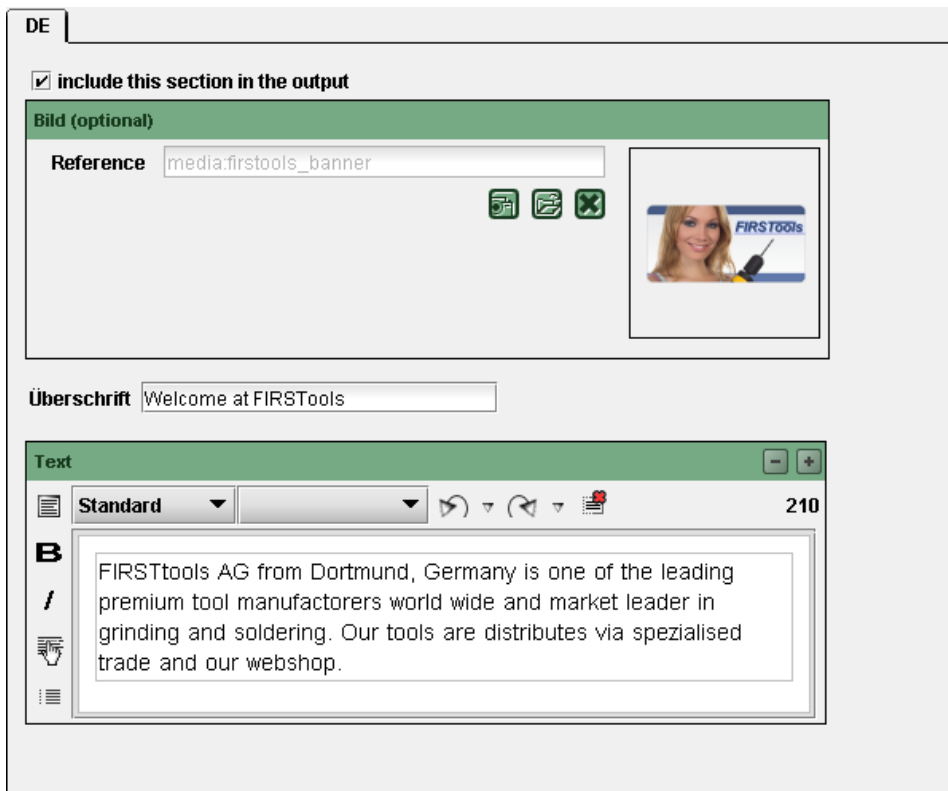
Figure 4-23: Section view

Generate this section in the output: This option is activated as a default, i.e. the section is included in the corresponding language in the next generation. If this option is deactivated then the section is not included when the project is generated.

In addition, there are various input components on this level which the template developer has earmarked for this page. Detailed documentation on the input components available is given in Chapter 10 page 249, general information about input components in the Chapters 11.5 to 11.8 from page 325.

5 Content Store of the JavaClient

The Content Store is used for entering and managing highly structured content. This includes e.g. product catalogues or address lists. These are not only highly structured but are also subject to frequent changes. Such data is usually kept in databases.

The tables of a databank can be integrated and maintained in the Content Store in the form of tables. They are then placed on a page as so-called data sources (content) via the Page and Site Store.

The following elements can be created within the Content Store:

- Folders for structuring the data sources (content).
- Tables
- Filtered Tables (see chapter 5.3.5ff. and 5.4.1.2 page 143 and 149)

5.1 General context menus in the Content Store
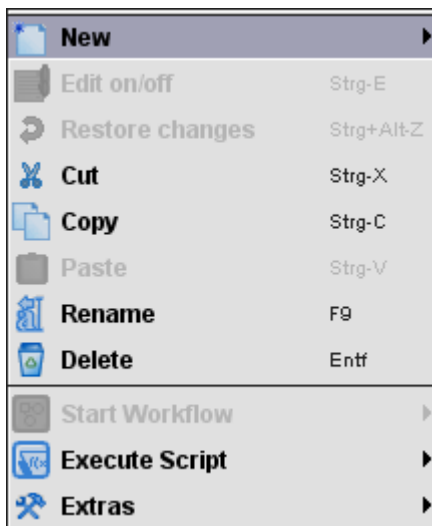
Figure 5-1: Context menu – General Content Store functions

The Content Store context menus are described in the following chapters:

For general information on handling context menus see Chapter 4.2 General context menus in the Page Store page 115.

5.1.1 New

The "New" context menu entry can be used to insert new objects into the project. The selection available depends on the object type on which the context menu was opened:

5.1.1.1 New: Root node and Folders
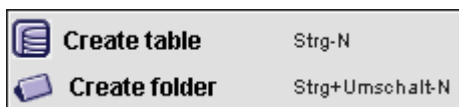
Figure 5-2: Selection at root node or folder level
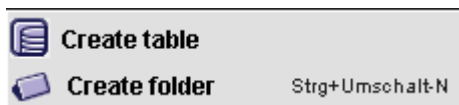
Create Table: This function can be used to insert a new table in the Content Store. A window opens in which a table template must be selected for the new table.

Figure 5-3: New – Create Table

The **Table** field is a combo box which can be used to select a table template from the database schema. The name of the selected table template automatically appears in the **Description** field. The description may be changed if necessary.

Create folder: It is advisable to file the individual tables in folders in the Content Store too for improved clarity. This function can be used to create a new folder, click and a window opens where you can enter the folder name.
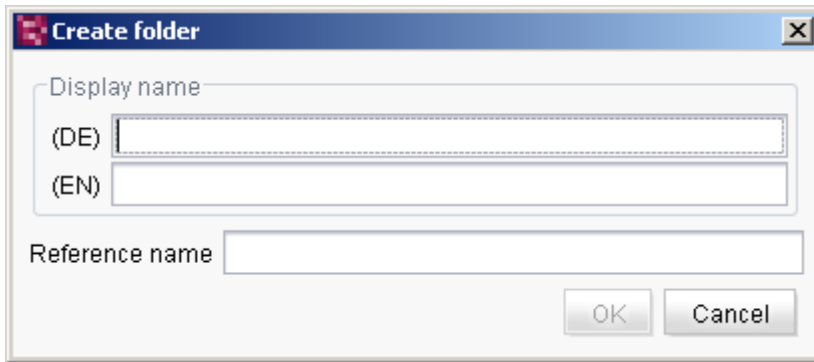
Figure 5-4: New – Create Folder (View version 4.2)

A **language-dependent display name** can be assigned to the new folder, for each editing language defined by the project administrator. Either the display names or the reference name are then displayed in the tree view, depending on the setting in the "Extras – Preferred Display Language" menu, from FirstSpirit Version 4.2 the "View – Preferred Display Language" menu (see Chapter 3.1.4.2 page 69). The Reference Name field is automatically filled with the value entered for the first display name by the editor, but can be changed (up to the initial creation of the object). The reference name may not contain any spaces, special characters or symbols. This is taken into account accordingly when the field is filled automatically.

> ! *The Reference Name field is only displayed if this setting was configured accordingly by the project administrator.*

> ! *In FirstSpirit Version 4.2R4 and higher, the administrator can define rules with which special characters in reference names are automatically transformed into valid characters. The conversion takes place directly during input when creating a FirstSpirit object or changing a reference name (context menu: "Extras" / "Change reference names").*
> *Special characters for which the administrator has not defined a replacement rule cannot be entered in the "Reference name" field.*

5.1.1.2   New: Table

Figure 5-5: Selection at table level

Create Table: This function can be used to insert a new table in the Content Store. A window opens in which a table template must be selected for the new table. (See also Figure 5-3: New – Create Table on Page 134.)

5.1.2   Lock/Unlock (Edit Mode On/Off)

This function is used to activate Edit Mode for a data source. At the same time, Edit mode is activated for the data set which is currently highlighted. Edit mode remains switched on even if other data sets of the same data source are selected. This makes it possible to edit several data sets in series.

Unlike in other stores, a data source will never be locked when switched to Edit mode so that different data sets can be created and edited in a data source at the same time. This is important for multi-user operations. If two users try to edit one data set simultaneously the so-called "optimistic lock method" will be applied (see Chapter 5.4.3 page 155, Figure 5-21).

Opening this function again deactivates Edit mode for the data source and the data set which is currently in Edit mode. Changes will be saved and the view mode will become active.

> *If a data source is to be refreshed (via F5 or the respective icon in the symbol bar (see Chapter 3.2.2 page 89)) for example to show modifications made by other users (e.g. creating new data sets), this data source must first be switched to view mode.*

5.1.3   Cut

This function is used to cut the current object and store it in the (temporary) clipboard. It can be inserted again in another position in the tree structure.

5.1.4 Copy

This function is used to generate a copy of the current object and store it in the (temporary) clipboard.

> **!** *At the level of a table, "Copy" copies the view of a database table only, not the selected data record.*

5.1.5 Insert

This function is used to insert the content of the (temporary) clipboard in the current position of the tree structure. This function is only active if there is data in the (temporary) clipboard which may be inserted in the current position.

5.1.6 Rename

This function can be used to change the name of the current object in the tree structure of the FirstSpirit JavaClient. After the function is executed a window opens with the object name to date; this can now be changed.

5.1.7 Delete (up to and including V4.2)

This function can be used to delete the current object from the tree structure of the FirstSpirit JavaClient. Accidental deletion is prevented by a confirmation prompt.

> **!** *A workflow, e.g. for deleting an object, can be tied to this function from FirstSpirit version 4.1 (see Chapter 3.2.9.7 page 101). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

> **!** *This function is deactivated from FirstSpirit Version 4.2R2.*

5.1.8 Starting workflow

All workflows defined in the permissions system for this node in the tree structure are listed under this menu item. Workflows are intended to help the editor carry out their tasks.

If the editor has e.g.: the permissions necessary to perform the "Request Release" workflow they can therefore release individual data records for publication on the website.

5.1.9 Execute Script

All scripts which can be opened in this position in the FirstSpirit JavaClient are listed under this menu item. Scripts enable pre-programmed actions or calculations to be executed. The release of individual data records, for example, is also possible by means of script (script can be made available on request).

5.2 Special context menus in the Content Store

5.2.1 Restore Deleted Objects (at table level)

The Restore Deleted Objects function can only be opened at table level. If a data record has been deleted from the table by mistake the deleted data record can be restored using this function. After clicking a window opens with the deleted data records.
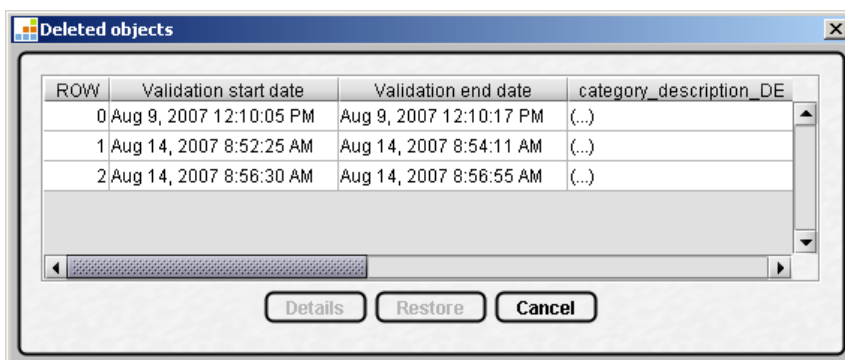
Figure 5-6: Deleted objects

The following information is given for each data record:

Row: The row in which the data record was located.

Validity Start: Date and time when the data record was created.

Validity End: Date and time when the data record was deleted.

Column Name (language-dependent): Name of the individual table columns. Brief information on the contents is displayed in the individual columns.

Deleted by: Name of the user who deleted the data record.

fs_id: The unique ID number of the deleted object

Released by: Name of the user who released the data record.

> ! *From FirstSpirit Version 4.1, the following window is displayed with the following information:*

Figure 5-7: Deleted objects

Revision: Version number of the deleted object

Date: Date and time when the data set was deleted.

Editor: Name of the user who deleted the data set.

Comment: Comment, assigned by the system, on the revision

ID: Unique ID number of the deleted object

Column name: Name of the individual table columns. Brief information on the

contents is displayed in the individual columns.

In the Page area, the arrow buttons can be used to page through the list of deleted objects. The buttons are only active if more than one page with deleted objects exists. In addition, the Objects per page input field can be used to set how many objects are to be displayed on each page. The input field also shows the user the page on which they currently are. This field can also be used to directly enter a page number. When the input is confirmed with Return, the focus then switches directly to the required page.

The Update button can be used to update the open list of deleted objects, if necessary, to the current status because new deleted data sets may have been added in the meantime.

Details Click this button to open a window with the complete contents of the listed versions.

Restore To restore, it is only necessary to select the required object and press the button.

On the right-hand side of the window a further context menu can be opened at each data record by right-clicking.
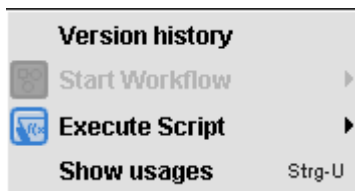
Figure 5-8: Context menu – Special functions at data record level

5.2.2 Version history (at data record level)

A window opens in which all the old versions of the current data record are listed. For precise documentation of version history in the Content Store see Chapter 11.12.4 page 407.

5.2.3 Start Workflow (at data record level)

If the editor has e.g.: the permissions necessary to perform the "Request Release" workflow they can therefore release individual data records for publication on the website.

5.2.4 Execute Script (at data record level)

The release of individual data records is, for example, possible by means of script too. (Script can be made available on request.)

5.2.5 Show Usages (at data record level)

This function can be used to determine at which position the current data record is referenced. A window opens in which all references are listed.

5.2.6 New bookmark (at data record level)

This function is only available from FirstSpirit Version 4.1.

Using this function bookmarks can be set on individual data records of the content store. for further informationen about the use of bookmarks see Chapter 3.2.3 page 89.

5.2.7 Display dependencies (at data record level, from V4.1)

Using this function the reference graph can be opened for the respective data record.

For a detailed documentation of the reference graph please refer to Chapter 11.13 page 413.

5.3 Administrative context menus in the Content Store
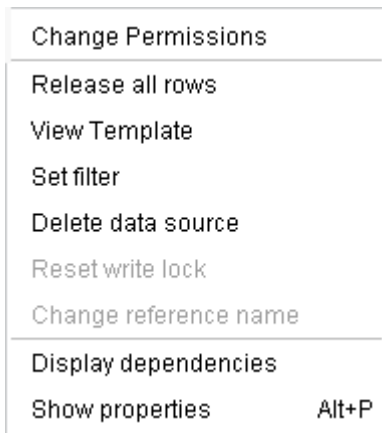
Figure 5-9: Context menu – Administrative functions at table level

5.3.1 Extras – Change Permissions

This function can be used to define the permissions for the current nodes in the tree structure.

Detailed documentation on the definition of permissions is given in Chapter 13.1.2 page 445.

5.3.2 Extras – Release All Rows (at table level)

This function can be used to simultaneously release all data records in a table, current filter settings are ignored.

5.3.3 Extras – Release displayed Rows (at table level, V4.2R4 and higher)

Call this function to release all data records, which are displayed depending on the current filter settings of the content (data source). All available filter settings are taken into account:

o   persistent filters via the "Extras" / "Set filter" context menu on the data source
o   temporary filters via the "Set filter" link above the table
o   temporary filters via the "Search" link above the "Data search" icon below a table,
o   temporary limitation of number of records displayed using the combobox above the table.

5.3.4 Extras – Go to Template (at table level)

If this function is executed the table template from the database schema used for the current table is displayed.

5.3.5 Extras – Set Filter (at table level, V4.1 and higher)

> **!** *This function is released for FirstSpirit Version 4.1 and higher only. Screenshots are therefore displayed in the new "LightGray" Look & Feel.*

This function is used to filter content (a data source) for data sets and can also set a sorting. This makes it easier to work with content which may possibly contains many similar data sets.

> **!** *To create filters (see Chapter 5.3.5.1 page 143) **before 4.2R2** the user requires "Create object" permission (see Chapter 13.1.4.5 page 458), **in 4.2R4 and higher** "Create folder" permission (see Chapter 13.1.4.6 page 458). To further edit filtered data records (see Chapter 5.3.5.2 page 145). on the other hand, "Change" permission is sufficient (see Chapter 13.1.4.4 page 457).*

5.3.5.1 Creating filters and sorting
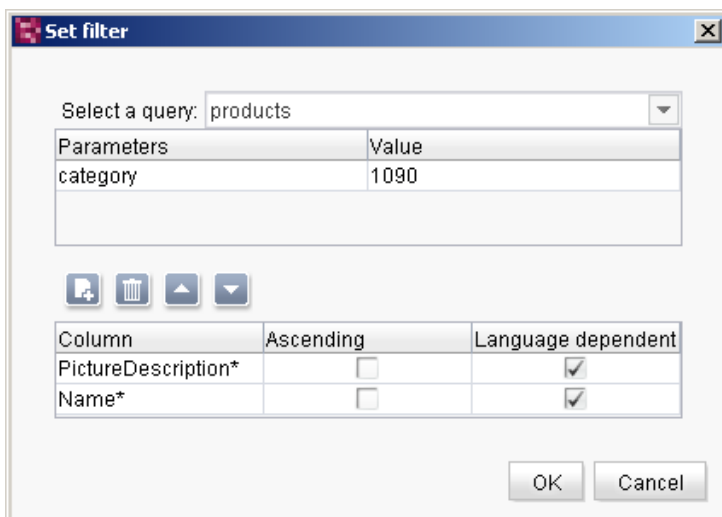
Figure 5-10: Filter data sets

After selecting this function, the dialog in Figure 5-10 in the top area can be used to select a database query from the Template Store, predefined by the project developer (cf. *FirstSpirit Manual for Developers*), for filtering.

After selecting a query, all parameters defined for the selected query are listed in the table underneath it. The given values of a parameter can be changed by double-clicking the relevant row.

The sorting of the filtered data sets can be defined in the bottom part of the dialog. Any table column can be used as the sort key. The output can be displayed either in ascending or in descending order and language-dependent or language-independent.

 Add Sort Rule: click this icon to use a selection box to specify another table column as the sort rule.

 Delete Sort Rule: click this icon to delete the activated sort rule.

 Move up/down: click these icons to move the selected sort rule up or down by one position in the list.

If several sort rules are specified, the top sort rule in the list is used for the sort order first. All entries for which this first sort rule is identical are then sorted by the second sort rule, whereby the overall first sort is retained. The procedure is the same for each additional sort rule.

Click OK to apply and save the filter and sort settings. The right-hand editing area now only shows the filtered data records with the required sorting. If a filter has been selected, the data source is displayed in the tree view with the additional name of the filter (database query) in brackets and the usual data source icon becomes . Filters and sorting and be changed again by opening the **Set filter** function from the context menu on the respective node.

> *In **V4.2R4** and higher, it is possible to limit the search result still further by selecting several filters. (Previously an existing filter was cancelled when another filter was selected.)*

5.3.5.2 Creating and editing data sets in a filter

If a new data set is created in filtered content , or an existing data set is changed so that it no (longer) satisfies the defined filter criteria, the following message is output after the data set is saved:
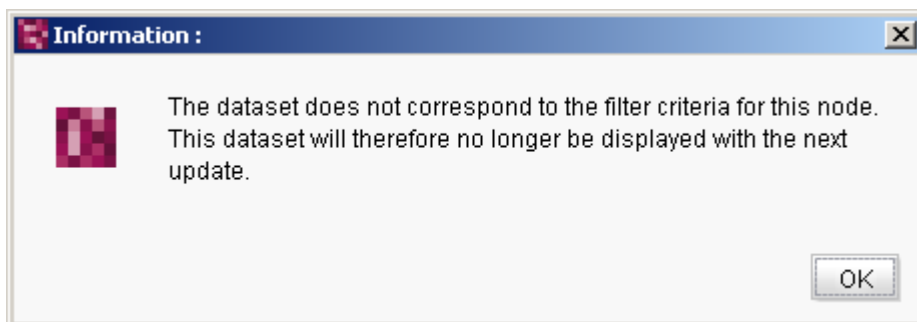
Figure 5-11: Data set does not satisfy filter criteria

The data set is no longer displayed following an update, e.g. using <F5> or a change to another node. The creation of filters in chapter 5.4.2 page 150 in more detail.

5.3.5.3 Search in filtered data sets

The icon can be used to search through filtered data sets. All search options described in Chapter 5.4.2 page 150 are available here.

5.3.6 Extras – Delete Data Source (at table level)

This can be used to delete the selected table – after confirming the confirmation prompt.

This function is available to administrators only.

5.3.7 Extras – Reset Write Lock

If a write lock exists for the selected node due to an active workflow the write lock can be cancelled using this function. (The write lock is indicated by italic lettering in the tree)

5.3.8 Extras – change reference name

Each FirstSpirit object has a reference name, which must be unique in each store. Each object can be identified by its reference name.

> **!** *Reference names are only shown in the project if the "Reference names" setting has been activated in the "Extras / Tree Display" menu, in FirstSpirit Version 4.2 and higher: "View – Preferred display language" (see Chapter 3.1.4.2 page 69).*

In general, reference names are issued automatically when objects are created and are based on the display name. These reference names can be subsequently changed using the "Change reference names" menu item. However, the user must have "Change" permission for the respective object. The reference name should only be changed if the object is not yet referenced in the project (e.g. a picture has not yet been used on a page or in a section of the Page Store or a page of the Page Store is not yet used in the Site Store ("page reference")), as otherwise the existing reference becomes invalid. The following dialog is displayed:

Change reference name

Reference name: mithras_home1

OK   Cancel

Figure 5-12: Change reference name

If the reference name is changed despite an existing reference, it may be necessary to restore the reference manually using new selection.

In FirstSpirit Version 4.2R4 and higher, depending on the project administrator's settings, the menu entry may be greyed out regardless of the "Change" permission; in this case the reference name cannot be changed.

5.3.9 Extras – Display dependencies (at table level, from V4.1)

Using this function the reference graph can be opened for the respective table.

For a detailed documentation of the reference graph please refer to Chapter 11.13 page 413.

5.3.10   Extras – Display Properties (at folder and table level, from V4.2)

This function can be used to show technical and editorial information on individual project content in a separate dialog and to collate it into a system report. It can also be opened using the keyboard shortcut Alt + P. The information can vary depending on the object type.
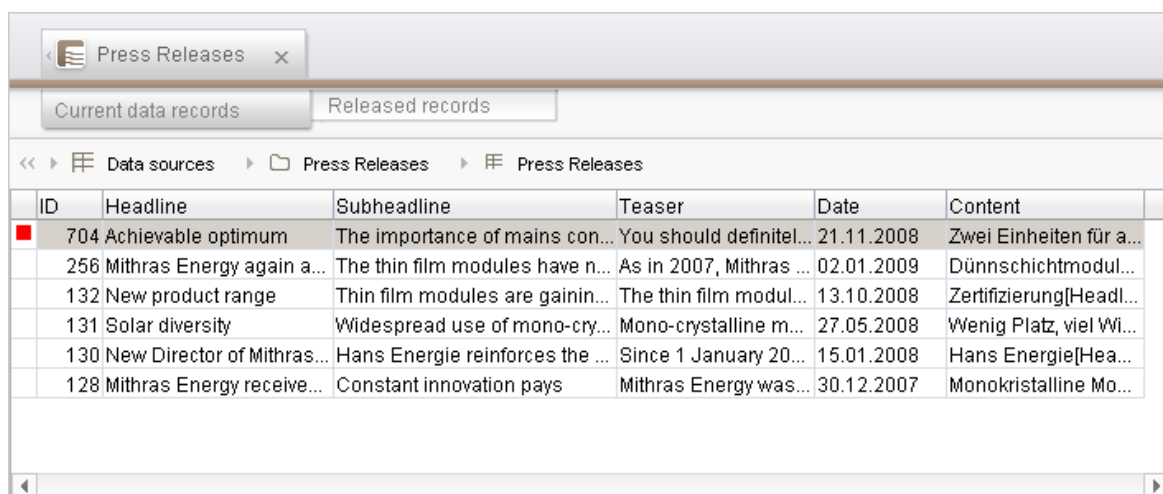
For further information about this function see Chapter 4.4.10 page 128.

5.4   Tables in the Content Store

After a new table has been inserted in the Content Store, data records can be entered in the right-hand side of the screen. If data records already exist they can also be further edited here (see Chapter 5.4.3 page 155).

5.4.1   Data Overview – Current Data Records

If content is selected in the tree structure, the overview of the current data reocrds is displayed:

Figure 5-13: Table overview – Current data records ("LightGray" Look & Feel)

All the available data records are listed in the table view. They are displayed on two tabs, separated according to current and released (see Chapter 5.4.4 page 158) data records. Click a column heading to sort the lists by the values in the respective column.

✓ 4.1   From FirstSpirit Version 4.1, current data records can be sorted by their modification and release status via the first column (modified and unreleased data
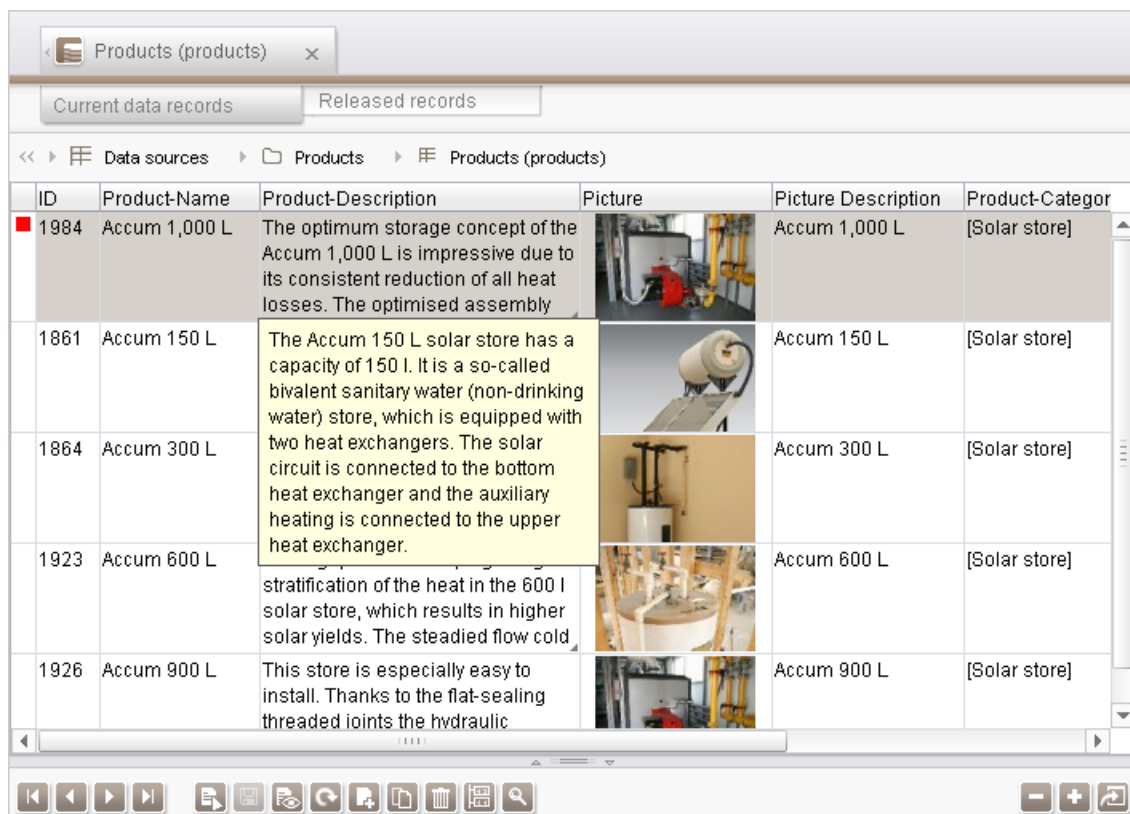
records are displayed at the top or bottom), the data records' ID is used as a second sort criterion.

In the area at the bottom of the dialogue box the data record selected in the list is displayed. As all the columns of a data record cannot always be displayed at the same time the order of the columns can be changed as required to enable the user to directly see the columns with the most important content. To change the order, click the column heading and drag it to the required position.

5.4.1.1 Multi-line display (from V4.1) ✓ 4.1

From First-Spirit Version 4.1, data records can be displayed with several lines:

Figure 5-14: Table overview – Multi-line display

Texts in a cell are automatically wrapped. If a cell contains more text than can currently be displayed, this is indicated by the ◢ icon in the bottom right-hand corner. If the mouse cursor is moved across the cell, the cell text is displayed in a tooltip. Pictures referenced within the data record are displayed 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 line height, width and order of the columns can be pre-set by the template developer. This default view can be changed manually; however, the default view is restored when the view is updated.

5.4.1.2   Limit the number of data records displayed in V4.2R2 and higher

From FirstSpirit Version 4.2R2, it is possible to set the number of data records to be displayed in data sources. This makes work easier, especially when working with data sources with lots of data records. This is done using a drop-down menu, located on the right above the data records (see Figure 5-15). The number of data records displayed can be limited to 100, 500, 1000 and 2000. The default limit is 500. Select the "Display all data sets" entry to display all data records. The user merely needs "Visible" rights to use this function.

The displayed data records can be further limited by using filtering (see Chapter 5.3.5 page 143) and the data search (see Chapter 5.4.2 from page 150). These functions already existed in earlier versions of FirstSpirit, but from FirstSpirit Version 4.2R2 they are made available directly above the data records. From FirstSpirit Version 4.2R2, the familiar filtering dialog can be opened by clicking the "Set filter" link, the search is opened using the "Search" link.

> **!** *The "Set filter" link is only active if the user has relevant permissions.*

All three options (filter, search, limit displayed data records) can be combined with each other. If filtering or a search is active, this is indicated by a ticked checkbox. In FirstSpirit Version 4.2R4 and higher these options are only temporarily valid, after <F5> or the corresponding icon in the toolbar is clicked (see chapter 3.2.2 page 89) on a data source in View mode, this object is once again displayed in its original view. Filter and Search mutually limit each other.

If the ID of a data record is known, it can be entered in the input field in the status line. After pressing the <RETURN> key, the required data record is displayed, provided any filtering that has been set allows this. As a default, the ID of the currently selected data record is displayed in this field.

Figure 5-15: Data source with search, filter and limitation

5.4.2   Data set selection

The following buttons are available for navigation in existing data records:

⏮ First: click this icon to display the first data record in the table.

◀ Previous: click this icon to display the previous data record in the table.

▶ Next: click this icon to display the next data record in the table.

⏭ Last: click this icon to display the last data record in the table.

🔍 Data search: click this icon to open a search window in which specific data

records can be found and picked out of the table with the respective status (current or released). There are different types of data search which are explained in the following chapters. The search results are displayed in the main window. The "Update" icon in the toolbar or <F5> must be pressed to display all the available data records.

### 5.4.2.1 Full text search (from V4.2)

Figure 5-16: Full text search

From FirstSpirit Version 4.2, the "Full Text Search" tab can be used to perform a search through all text content in a table, including texts deposited in input components of the type DOM Editor. To do this, a search term is entered in the "Search Text" field. 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.
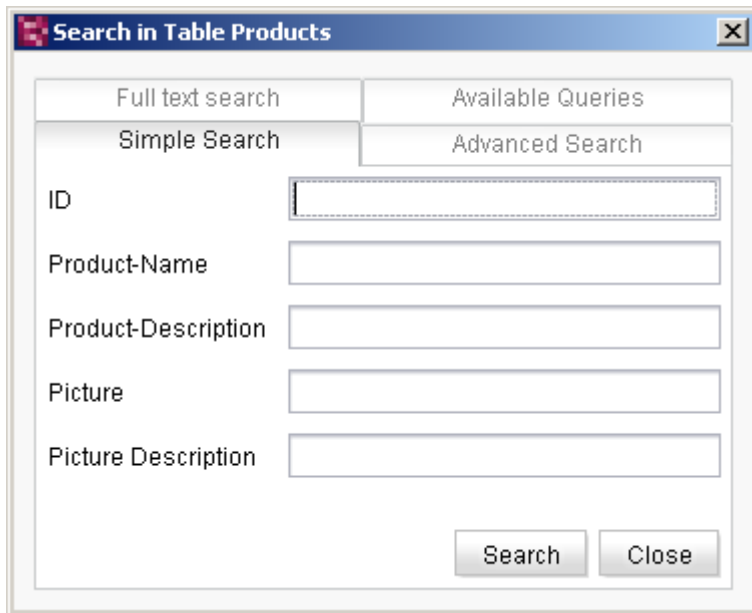
5.4.2.2 Available Queries
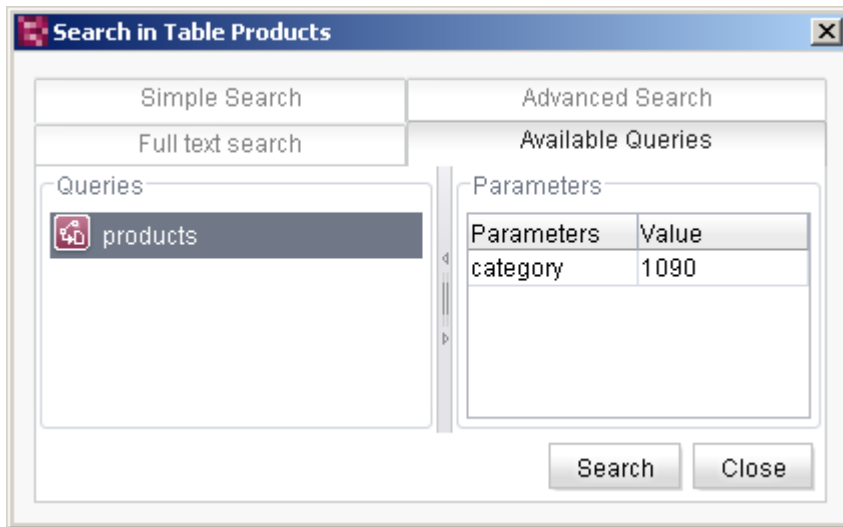
Figure 5-17: Available Queries

Prepared standard search queries can be used in the "Existing Query" tab. These queries have been created by the project developers to make subsequent searching for data records easier. Only one of the existing search queries has to be selected and the **Search** button activated. The search results appear in the main window.

5.4.2.3 Simple Search

Figure 5-18: Simple Search

The "Simple Search" tab can be used to search for specific content in one of the input fields of a data record. The required search term must be entered in the relevant input field and the **Search** button activated. If "ID" is searched for, only the precise ID given in the search field is sought. A full search is performed for searches in other input fields of the data records; however, these searches are *case sensitive*. References contained in the data records are also searched through, e.g. references to media, page references, etc. Searches in input fields in which, for example, pictures are references, search through the respective reference names of the pictures. Terms can also be entered in several fields to further limit the search. The search results appear in the main window.

5.4.2.4 Advanced Search
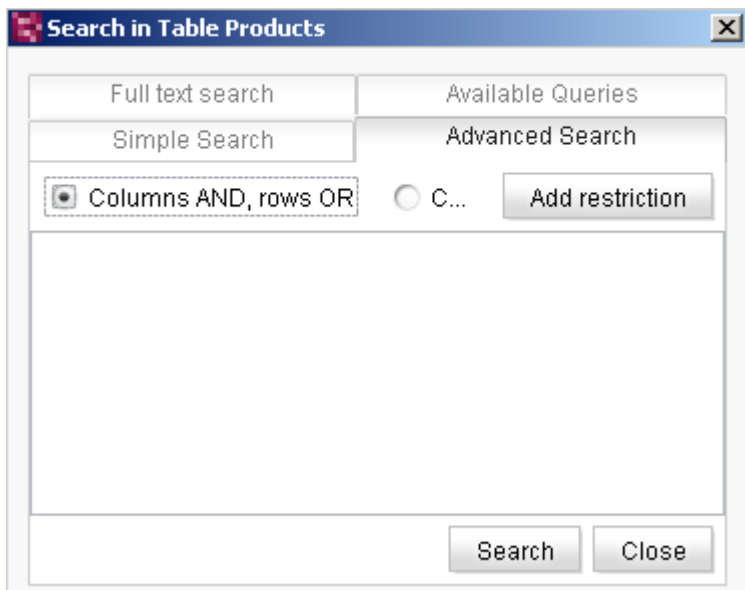
Figure 5-19: Advanced Search

The "Advanced Search" tab can be used on the one hand to create your own queries similar to the "Existing Search Queries" (see Chapter 5.4.2.1 page 151). On the other hand, the "Advanced Search" can also be used to search through fields which are not filled by the direct input. The contents of these fields are managed in a table and are integrated in another table. This is done using the so-called "foreign keys". A precise description of the Advanced Search is given in the "Query" section of the chapter on Database Schemata in the *Documentation for Developers*" manual.

5.4.2.5 Display of search results from V4.1 ✓ 4.1

If a data search is performed from FirstSpirit Version 4.1, the results are displayed in the overview. The search activated is displayed above the list, in the example in Figure 5-20 for example the existing search query *products*:

Figure 5-20: Search in filtered content

Search: xyz checkbox: This checkbox can be used to quickly activated and deactivate the search so that only the data records to which the search criteria apply or all data records can be displayed.

, click the icon to delete the search query and all data records of the filtered content are displayed again.

5.4.3   Data entry

The bottom area for data entry is displayed by clicking the icons: , , , or (from FirstSpirit Version 4.2). The fields of the data records can be filled or changed here. The area can be made larger or smaller using the splitter bar between the overview and editing area; to do this, move the bar with the icon bar up or down, or use the +/- icons. Double-click an existing data record in the data overview to open the detailed view in a separate window. The same icons for data record navigation and editing are available in the detailed view.

From FirstSpirit Version 4.2, during editing the currently edited data record is displayed within the overview with stronger highlighting and in bold. In this way, the editor can recognise at a glance which data is currently being changed If the editor selects a new data record in the overview, the focus in the bottom editing area changes and the now current data record is highlighted bold in the overview.

 Edit data record: click this icon to change an existing data record. Similar to the other stores, edit mode is switched on as a result and the icon is displayed "pressed in". A renewed click on the icon switches edit mode back off again, any changes

made in the meantime are saved. If edit mode is switched on for a data record, it remains switched on even if other data records of the same content are selected. This makes it possible to edit several data records in series.

> **!** *If a data record which is in edit mode and has been changed is quit, e.g. by selecting another data record in the data overview or using the page icons, by closing the detailed view or the project, a confirmation prompt appears, asking whether the previously made changes are to be saved or not.*

💾 Save: click this icon to save the changes made to an existing data record and the values of a new data record. Edit mode is not quit. The icon is only active if changes have been made to the data record.

> **!** *Unlike elsewhere, the data nodes are not blocked against data entry. Therefore, various data records can be simultaneously created and edited within a node.* **Parallel working within a database view** *is advantageous for multi-user operation, but can lead to conflicts if two editors try to change the same data record at the same time.*
> *In this case, a so-called* **"optimistic lock method"** *is applied. In this case, only the changes of the editor who was the first to save the data record are accepted initially. If the second editor now tries to save the changes to the data record, the following message is displayed to them: "The record has been changed by another editor. Your changes could not be saved!"*
> *The editor can then update the data record to display the changes of the second editor or save their own changes despite this and therefore overwrite the changes of the other editor.*

Figure 5-21: Conflict in the event of parallel editing of a data record

👁 Preview: click this icon to display a preview of the selected data record.

> ! *Restrictions for the output of data records which have been made in the Site Store will not be taken into account for the preview of single data records in the Content Store.*
> *The preview of a single data record always can be displayed with the respective Preview page.*

> ! *From FirstSpirit Version 4.1, any changes not yet saved are not saved when the preview is opened using this icon. If, on the other hand, a preview is opened using the "Preview" icon of the JavaClient toolbar, unsaved changes are automatically saved first*

Update: if changes are made by another editor to a data record displayed in the bottom part of the window or in the detailed view, these changes are not automatically updated. Click this button to update the data record view.

New entry: this icon is used to create a new empty data record.

New Entry (adopt values): this icon is used to create a new data record as a copy of the selected data record. The values of the copied data record are adopted for the new data record. This icon can be inactive, depending on the configuration by the template developer.

Delete: click this icon to delete the entry currently displayed from the current (unreleased) status.

> ! *If a data record is to be completely deleted from the content, it must also be deleted with released status (see Chapter 5.4.4 page 158). All deleted data records can be restored again at any time using the "Restore Deleted Objects" function in the context menu of the respective content.*

Version management: click this icon to open a window listing all versions of the current data record. For precise documentation of the version history in the Content Store, see Chapter 11.12.4 from page 407.

Data search: click this icon to open a search window which can be used to

search through the table of the current data records. The different types of data search are explained in Chapters 5.4.2.1 to 5.4.2.4, from page 151.

Minimise / Maximise: click this icon to reduce or increase the size of the editing area.

(from FirstSpirit Version 4.2): click this icon to open the selected data record in a separate editing window.

> *Further functions can be called using the context menu of a data record (see Chapter 5.2.2 ff., from page 140).*

5.4.4 Data overview – Released records

If the project uses the release option, released data records are displayed on a separate tab:

Figure 5-22: Table overview – Released data records

The display of the table overview and the navigation or data record selection corresponds to that of the current data records (see Chapter 5.4.1 page 147).

> ![] *From FirstSpirit Version 4.1, released data records can be sorted by the time of the last release via the first column In this way, the most recently released data records can be displayed at the top of the list with a click.*

![] Update: this icon can be used to update the data overview of the released data records if necessary, because new data records may have been released in the meantime.

![] Delete: data records which only exist with released status are identified by grey marking in the first column. Click this icon to delete the selected data record with released status. All deleted data records can be restored again at any time using the "Restore Deleted Objects" function in the context menu of the respective content.

![] Data search: click this icon to open a search window which can be used to search through the table of released data records. The different types of data search are explained in Chapters 5.4.2.1 to 5.4.2.4, from page 151.

> ![] *Further functions can be called using the context menu of a data record (see Chapter 5.2.2 ff., from page 140).*

6   Media Store of the JavaClient

The Media Store is used to manage all the media used. These can be images in any graphic format which are to appear on one or several pages of the website. They can include PDF files to be offered for downloading, sound files which are played back when certain links are clicked, etc.

The following elements can be created within the Media Store:

- Folders for structuring the media.
- Language-independent pictures
- Language-independent files
- Language-dependent pictures
- Language-dependent files

> *In FirstSpirit Version 4.2R4 and higher, specific icons are displayed for files, which are based on the file type. In this way, the data type is immediately apparent in the tree structure, without having to change to the detailed view first.*

6.1   "Drag & Drop" in the Media Store

> *The drag & drop functions up to FirstSpirit version 4.1 including are described in the following. For a detailed documentation of the enhanced drag & drop functions from FirstSpirit version 4.2 on see Chapter 11.4 page 318.*

Folders and media can be moved within the Media Store with the help of the mouse using "Drag & Drop" (identified by a small rectangle on the mouse pointer).

> ❗ *If the function "Confirm move operations" in the menu "Extras" is active, for each move a security dialog box must be confirmed (see Chapter 3.1.5.2 page 78).*

As an alternative to creating media via the context menu (Chapter 6.2.1 page 162), media from the directory structure of the workstation computer can be added to the Media Store or into Media Store folders by means of "Drag & Drop". The file name is automatically adopted, the display name must be manually assigned afterwards.

> ❗ *If files are added to the Media Store by means of "Drag & Drop" from the directory structure of the workstation computer, which are not allowed due to the configuration in the project properties (see also 6.2.1.1 page 162, notes on media restrictions from FirstSpirit Version 4.1), a corresponding message is displayed. The files are not created in the Media Store.*

> ❗ *If Drag & Drop (move, copy) is used on nodes within or into the Media Store, an error message appears "Unable to perform this action (insufficient permissions)!", if the user does not have the permissions to perform this action!*
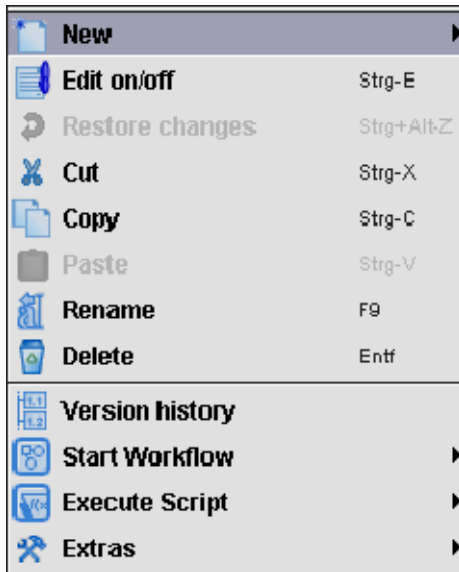
6.2 General Media Store context menus

Figure 6-1: Content menu – General Media Store functions

The Media Store context menus are described in the following chapters:

For general information on handling context menus see Chapter 4.2 General context menus in the Page Store page 115.

6.2.1 New

The "New" context menu entry can be used to insert new objects into the project. The selection available depends on the object type on which the context menu was opened:

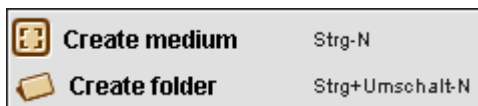#### 6.2.1.1 New: Root node and Folders

Figure 6-2: Selection at root node or folder level

New – Create new medium: This context menu entry can be used to add a new medium to the media management. Click and a dialog box opens in which the details for the new media can be entered.
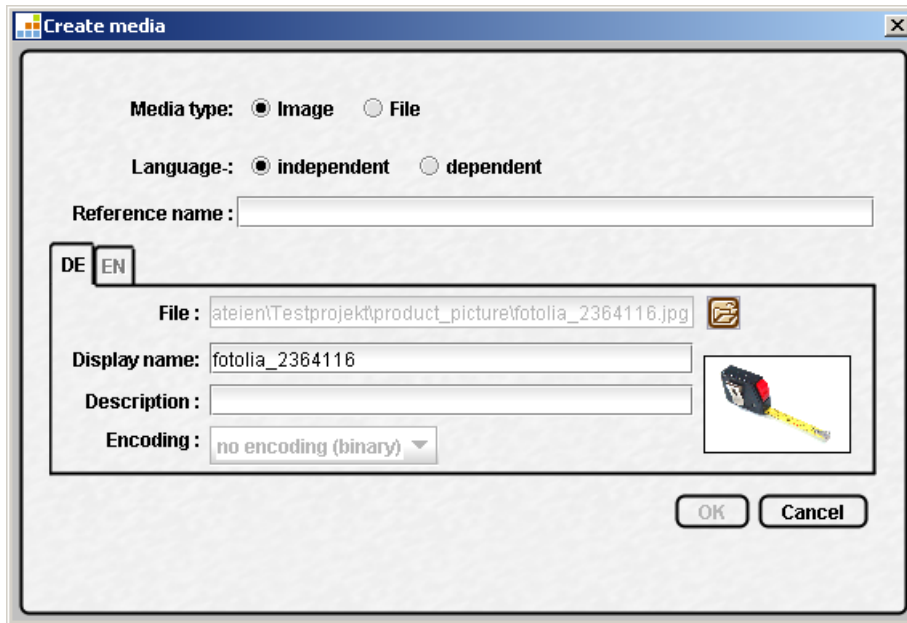
Figure 6-3: New – Create new medium

In the top part of the dialog box, in the **Media Type** line, you can select whether the medium is an image or a file. Use the **Language** line to define whether the new medium is to be created language-dependent or language-independent. Also, a unique name for the new media must be entered in the **Reference Name** field.

A **language-dependent display name** can be assigned to the new folder, for each editing language defined by the project administrator. Either the display names or the reference name are then displayed in the tree view, depending on the setting in the "Extras – Preferred Display Language" menu, from FirstSpirit Version 4.2 the "View – Preferred Display Language" menu (see Chapter 3.1.4.2 page 69). The Reference Name field is automatically filled with the value entered for the first display name by the editor, but can be changed (up to the initial creation of the object). The reference name may not contain any spaces, special characters or symbols. This is taken into account accordingly when the field is filled automatically.

> ❗ *The Reference Name field is only displayed if this setting was configured accordingly by the project administrator.*

In the bottom part of the dialog box you can select a medium for each language of language-dependent media, otherwise select a medium for the master language only. Click the 📂 icon in the **File** field to select the required image or file from the

directory structure of the desktop computer.

From FirstSpirit Version 4.1 the selection of media via the file selection dialog can be restricted to files of specific size and format according to the settings in the project configuration (see *FirstSpirit Manual for Administrators*).

> ! *The functionality of the media restriction is only released from FirstSpirit Version 4.1. For this reason, the screenshots are in the new Look & Feel "LightGray". The representation in the Look & Feel "Classic" can slightly differ.*
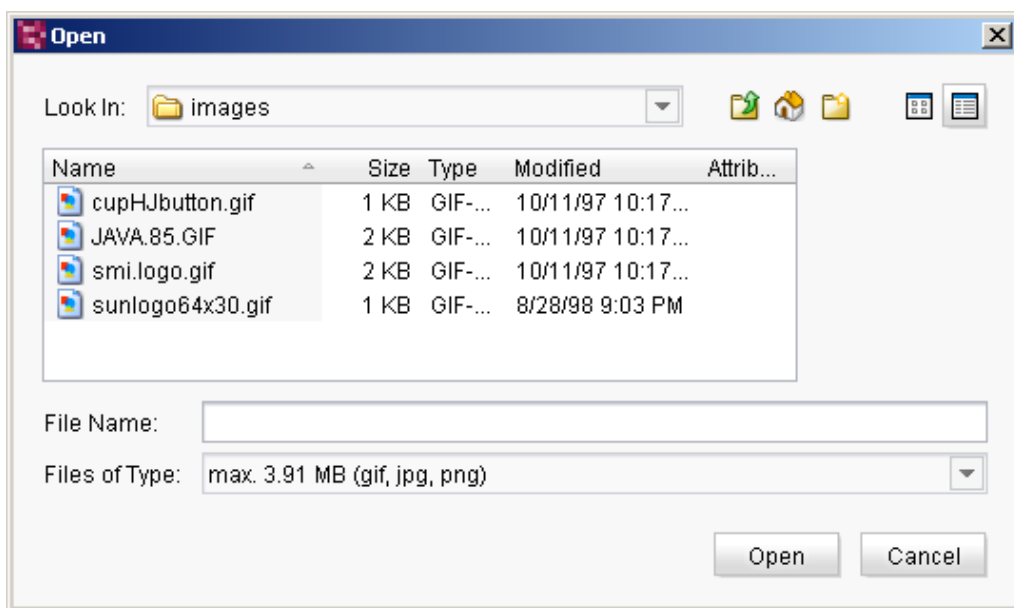

Figure 6-4: File selection dialog with media restriction

Media which is larger than the maximum size defined in the project properties and/or has a file name extension which is not allowed, is not made available to choose from in the file selection dialog. The restrictions are displayed again in the "Files of Type:" field. In the example in Figure 6-4, files with file name extensions "jpg", "png" and "gif" up to a size of 3.91 MB may be selected and uploaded. The settings of the respective project apply, i.e. if media are uploaded into a remote project (see Chapter 11.14 page 416), the media restrictions of the remote project apply and if media are uploaded into the target project, the media restrictions of the target project apply. Click "Open" to copy the selected medium into the dialog in Figure 6-3.

> **!** *A corresponding warning appears if files from the workstation computer's directory structure which are not allowed due on the basis of the media restrictions are added to the Media Store by means of "Drag & Drop". The files are not created in the Media Store.*

An explanatory text for the image can be entered in the **Description** field; this is used in the Media Store only. When adding files, use the **Encoding** field to set the standard by which the original file is to be converged. (More precise information on the individual character sets are given in the *FirstSpirit Manual for Administrators*.)

Click on the button "OK" to take over the medium to the Media Store. The dialog will close.

New – Create Folder: For the Media Store, as elsewhere, it is advisable not to store all the media in one directory but to set up a meaningful folder structure to make it easy to find individual media again. Depending on the number of media, you can also create sub-folders in these folders which, e.g. reflect the structure from the Page and Site Stores. This function can be used to insert a new folder in the tree structure. Click to open a dialog box in which you can enter the folder name.
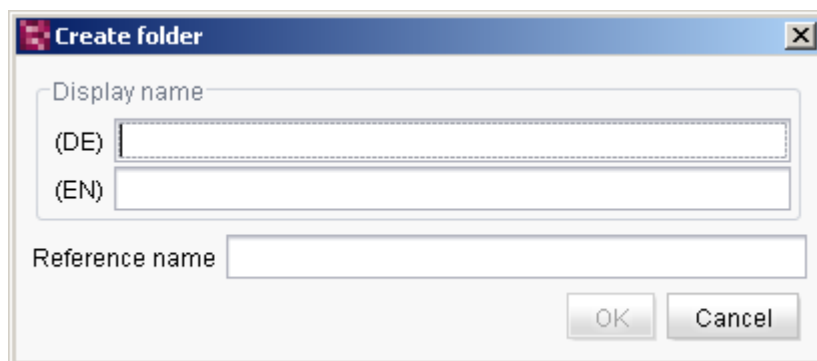

Figure 6-5: New – Create Folder

A **language-dependent display name** can be assigned to the new folder, for each editing language defined by the project administrator. Either the display names or the reference name are then displayed in the tree view, depending on the setting in the "Extras – Preferred Display Language" menu, from FirstSpirit Version 4.2 the "View – Preferred Display Language" menu (see Chapter 3.1.4.2 page 69). The Reference Name field is automatically filled with the value entered for the first display name by the editor, but can be changed (up to the initial creation of the object). The reference name may not contain any spaces, special characters or symbols. This is taken into account accordingly when the field is filled automatically.

> ⚠️ *The Reference Name field is only displayed if this setting was configured accordingly by the project administrator.*
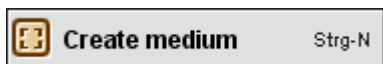
6.2.1.2 New: Media

Figure 6-6: Selection at media level

New – Create new medium: This context menu entry can be used to add a new medium to the Media Store. Click and a dialog box opens in which the details for the new media can be entered (see Figure 6-3: New – Create new medium on Page 163).

6.2.2 Lock/Unlock (Edit Mode On/Off)

This function is used to activate Edit Mode for the selected node. No other editor can then make changes to this object because it is locked.

Opening this function again deactivates edit mode (unlocks the object), i.e. View Mode is reactivated.

6.2.3 Undo

This function can be used to undo changes made during the current editing process and which have not yet been saved.

6.2.4 Cut

This function is used to cut the current object and store it in the (temporary) clipboard. It can be inserted again in another position in the tree structure.

6.2.5 Copy

This function is used to generate a copy of the current object and store it in the (temporary) clipboard. Copies can be generated from pictures and files.

6.2.6 Insert

This function is used to insert the content of the (temporary) clipboard in the current position of the tree structure. This function is only active if there is data in the (temporary) clipboard which may be inserted in the current position.

6.2.7 Rename

This function can be used to change the name of the current object in the tree structure of the FirstSpirit JavaClient. After the function is executed a window opens with the object name to date; this can now be changed.

6.2.8 Delete

This function can be used to delete the current object from the tree structure of the FirstSpirit JavaClient. Accidental deletion is prevented by a confirmation prompt.

> **!** *A workflow, e.g. for deleting an object, can be tied to this function from FirstSpirit version 4.1 (see Chapter 3.2.9.7 page 101). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

6.2.9 Version history

A window opens in which all the old versions of the current object are listed. For precise documentation of version management at media level in the Media Store see Chapter 11.12.5.1 page 409.

6.2.10 Workflow

If workflow is not yet active for the selected object, all workflows defined in the permissions system for these nodes in the tree structure are listed under this menu item. The required workflow can be started under this menu item.

If a workflow is already active for the selected object it can be switched to another workflow action/state under this menu item.

Detailed documentation of workflows is given in Chapter 12 page 422.

6.2.11 Execute Script

All scripts which can be opened in this position in the FirstSpirit JavaClient are listed under this menu item. Scripts enable pre-programmed actions or calculations to be executed.

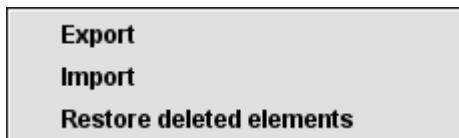## 6.3 Special Media Store context menus in the Media Store

Export
Import
Restore deleted elements

Figure 6-7: Context menu – Special functions at folder level

6.3.1 Export (at folder and media level)

This feature is only available to administrators.

This function can be used to export the selected folder or the selected medium with all the necessary information to the hard disk, from where it can later be imported again into another project. A window opens with the file structure of the computer to enable you to specify a suitable storage location for the export.

> **!** *This function can be used to export the selected folder or the selected medium with all the necessary information to the hard disk, from where it can later be imported again into another project. A window opens with the file structure of the computer to enable you to specify a suitable storage location for the export.*

6.3.2 Import (at root and folder level)

This function can be used to import an export file with all the necessary information back into the project. A window opens with the file structure of the computer enabling you to select the required export file.

6.3.3 Restore Deleted Objects (at root and folder level)

The Restore Deleted Objects function can be opened both at root and at folder level. If a folder or a medium has been accidentally deleted from the tree structure this

function can be used to restore the deleted object. After clicking a window opens with the deleted objects.
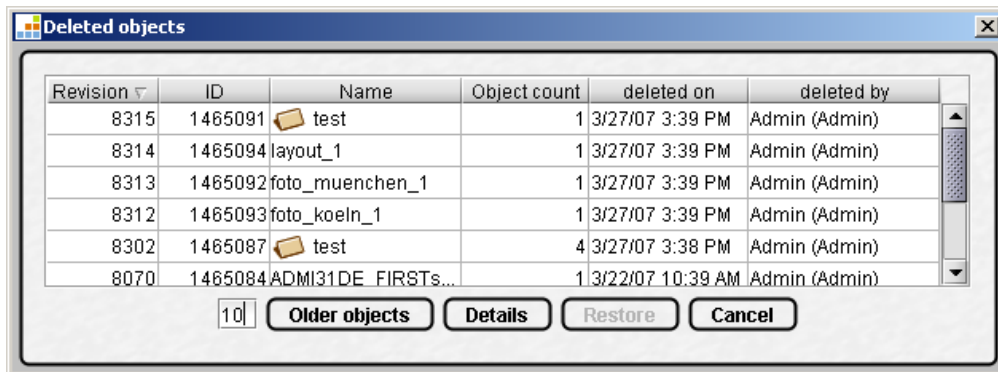

Figure 6-8: Deleted objects

All objects for which a backup exists are displayed at root level, while at folder level only the objects located underneath this folder are displayed. The following information is given for each object:

Revision: Version number of the deleted object.

ID: The unique ID number of the deleted object

Name: The reference name of the deleted object.

Number of objects: The number of objects located in the tree structure below the deleted object. These hierarchically lower level objects are also inserted again by the restore function.

Deleted on: Date and time when the object was deleted.

Deleted by: Name of the user who deleted the object.

Restore To restore, it is only necessary to select the required object and press the button.

The restored objects are, where possible, inserted in the same position in which were located before being deleted. However, if e.g. the folder in which a page to be restored has also been deleted (and is to remain deleted), this page is inserted directly below the tree node at which this function was opened.

6.4   Administrative context menus in the Media Store

Figure 6-9: Context menu – Administrative functions at media level

6.4.1   Extras – Release

This function can be used to directly release the current object or a specific release option can be used.

Detailed documentation on the specific release options is given in Chapter 12.4 page 433.

6.4.1.1   Extras – Release – All Objects (at folder level)

This function can be used to release all the Media Store objects in one go, including objects already released beforehand.

Media being edited when this function is executed are ignored. If there are very many media long waiting times can result when the media are released; an hourglass appears which means that further actions cannot be carried out on client.

6.4.1.2   Extras – Release – Release (at folder level)

This function can be used to release all media which are in the current folder, i.e. the media are taken into account in the next generation.

6.4.2   Extras – Delete Metadata (root, folder and media level)

Execute this function to delete all metadata defined for the current level in the tree structure.

6.4.3 Extras – Change Permissions

This function can be used to define the permissions for the current nodes in the tree structure.

Detailed documentation on the definition of permissions is given in Chapter 13.1.2 page 445.

6.4.4 Extras – Reset Write Lock

If a write lock exists for the selected node due to an active workflow the write lock can be cancelled using this function. (The write lock is indicated by italic lettering in the tree)

6.4.5 Extras – Show Usages

This function can be used to show all places in which the selected medium is being used. A window opens in which the used positions are listed.

6.4.6 Extras – Display Properties (from V4.2)

This function can be used to show technical and editorial information on individual project content in a separate dialog and to collate it into a system report. It can also be opened using the keyboard shortcut Alt + P. The information can vary depending on the object type.

For further information about this function see Chapter 4.4.10 page 128.

6.4.7 Extras – Make Medium Language Dependent

This function can be used to subsequently transform a language-independent medium into a language-dependent medium or vice-versa.

To convert a language-independent medium into a language-dependent medium it is necessary to give the language for which the existing medium is to be used. If a language-dependent medium is to be converted into a language-independent medium then it is necessary to give the language for which the medium is to be retained, if it already exists for different languages.

The permissions set for the old medium are imported into the new medium!

6.4.8   Extras – Cancel Editing

Using this function the editing mode can be determined, without accepting changes which have been not saved yet.

6.4.9   Extras– change reference name

Each FirstSpirit object has a reference name, which must be unique in each store. Each object can be identified by its reference name.

> ! *Reference names are only shown in the project if the "Reference names" setting has been activated in the "Extras / Tree Display" menu, in FirstSpirit Version 4.2 and higher: "View – Preferred display language" (see Chapter 3.1.4.2 page 69).*

In general, reference names are issued automatically when objects are created and are based on the display name. These reference names can be subsequently changed using the "Change reference names" menu item. However, the user must have "Change" permission for the respective object. The reference name should only be changed if the object is not yet referenced in the project (e.g. a picture has not yet been used on a page or in a section of the Page Store or a page of the Page Store is not yet used in the Site Store ("page reference")), as otherwise the existing reference becomes invalid. The following dialog is displayed:

Figure 6-10: Change reference name

If the reference name is changed despite an existing reference, it may be necessary to restore the reference manually using new selection.

In FirstSpirit Version 4.2R4 and higher, depending on the project administrator's settings, the menu entry may be greyed out regardless of the "Change" permission; in this case the reference name cannot be changed.

6.4.10 Extras – Display dependencies (at Media level, from V4.1)

Using this function the reference graph can be opened for the respective medium.

For a detailed documentation of the reference graph please refer to Chapter 11.13 page 413.

6.5 Settings at folder level

The same view and setting options are available for the root of the Media Store, as in the folders of the Media Store.

6.5.1 Overview tab
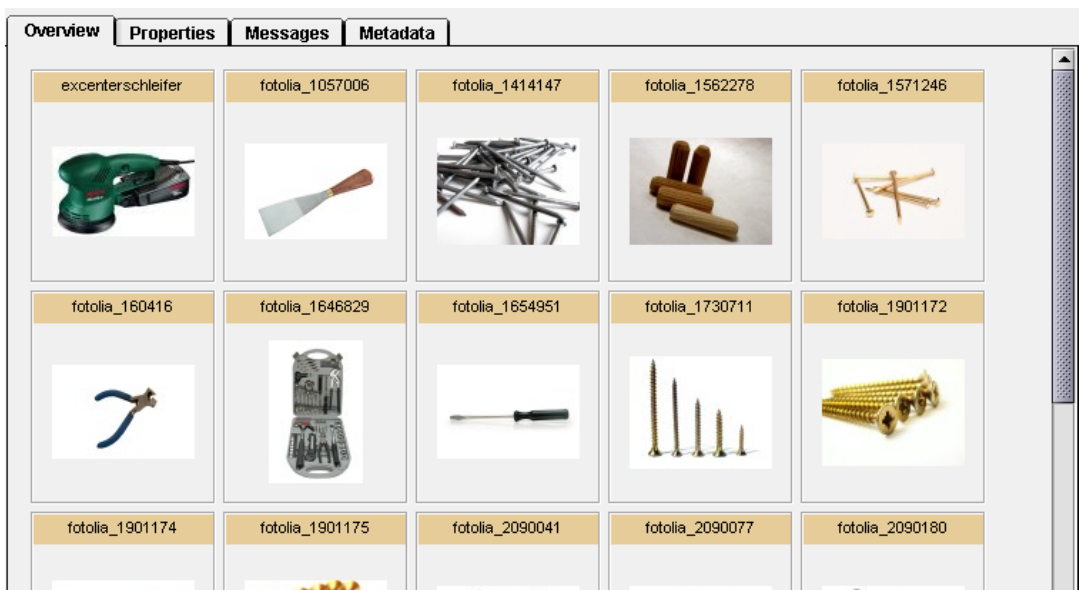
All folders and media located in this folder are displayed in the Overview tab.

Figure 6-11: Folders view – Overview tab

In Version 4.2R4 and higher, specific icons based on the file type are also shown at folder level in the "Overview" tab for files.
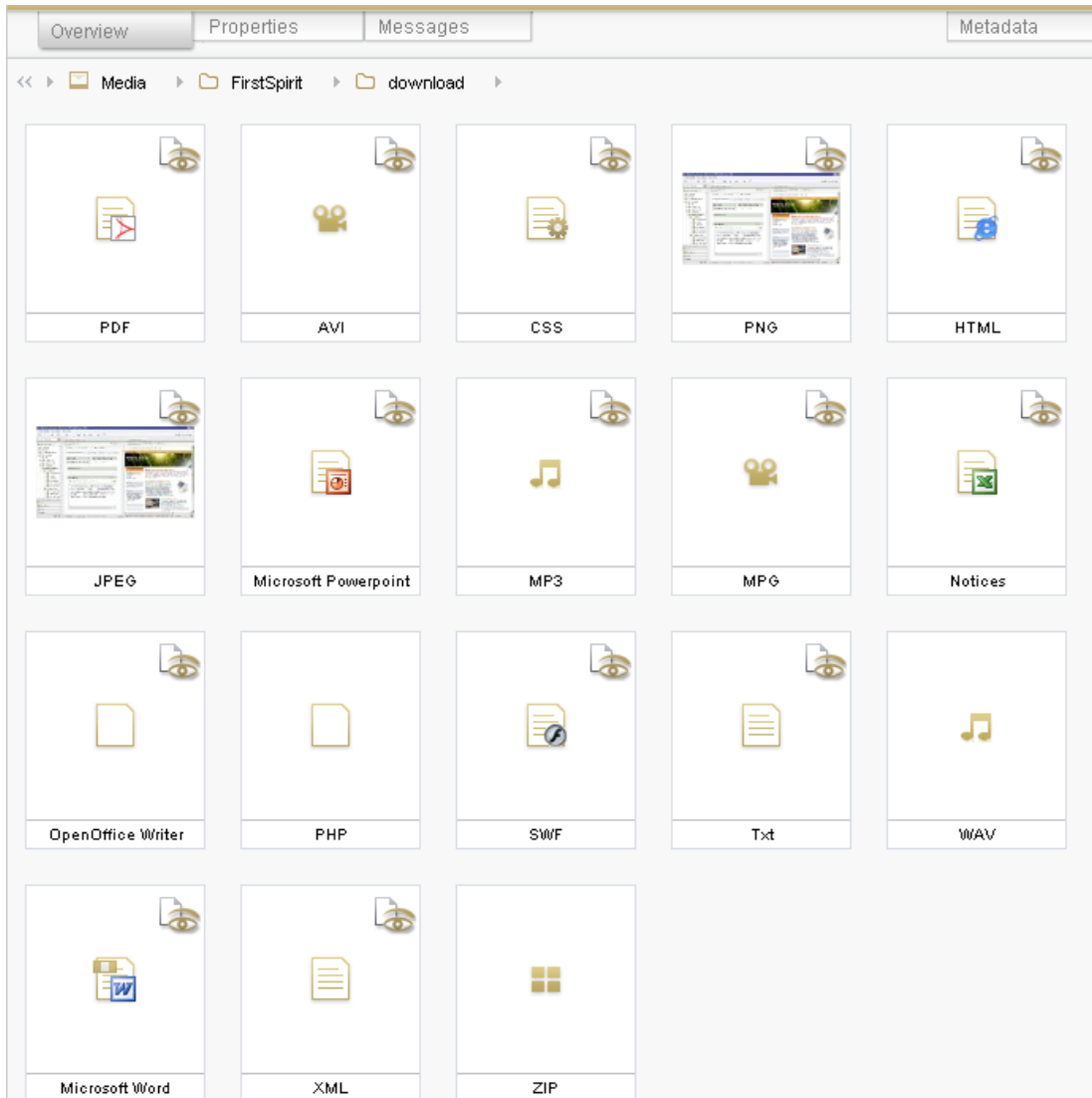
Figure 6-12: Folders view – Overview tab Version 4.2R4

File types for which no separate icon is used are displayed with the ☐ icon, language dependent files are especially identified: ↗

If the Integrated preview is enabled for media (see Chapter 3.1.4.4 page 72), the icon indicates that the file concerned can be displayed in the Integrated preview (see Chapter 1.3.2.6 page 35). The medium will open in the integrated preview by one click on this icon.

6.5.2 Properties tab

Various settings can be specified in the Properties tab.
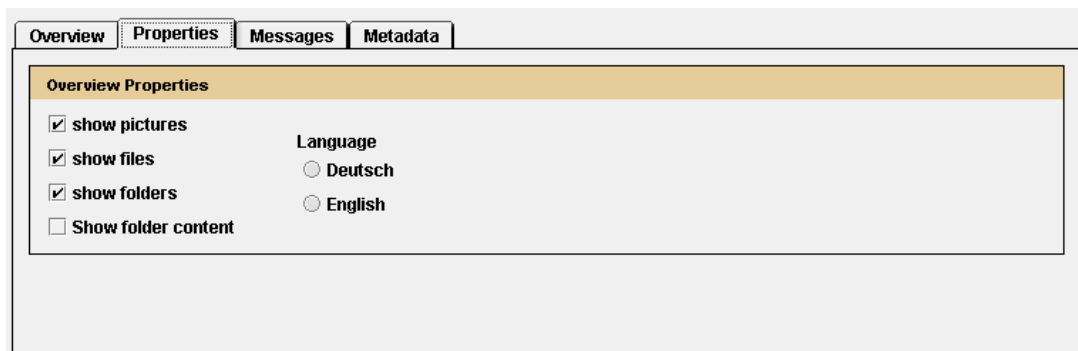
Figure 6-13: Folders view – Properties tab

6.5.2.1 Media Properties

Copy all in this folder on generation: Activate this option and all the media in the folder are copied on generation and not only the media which are already referenced. For pictures rendered with the original resolutions only to date this means that during the next generation all other resolutions will be rendered and saved.

> *This option is available from FirstSpirit version 4.1 on only in the "compatibility mode 3.1" (see FirstSpirit Manual for Administrators, Chapter 7.4.2) and will be dropped with FirstSpirit version 4.2 (cf. FirstSpirit Release Notes for version 4.1).*

6.5.2.2 Overview Properties

In this area you can set which contents are to be displayed in the Overview tab.

Display Pictures: All pictures in this folder are displayed.

Display Files: All files in this folder are displayed.

Display Folders: All sub-folders of this folder are displayed.

Show Folder Content: The respective contents of the sub-folders are also

displayed.

Language: In projects with several project languages, these radio buttons can be used for language-dependent pictures to select the language in which the corresponding thumbnails are to be displayed.

> ! *The folder does not have to be blocked to change the settings; however, the user must have "Read" permission.*

> ! *The folder does not have to be blocked to change the settings; however, the user must have "Change" permission.*

6.5.3   Messages tab

The Messages tab contains a message board.

Detailed documentation on use of message boards is given in Chapter 11.2 "The Message Board" page 316.

6.6   Settings at Media level

6.6.1   Pictures

The setting options for language-dependent and language-independent pictures are identical. For language-dependent pictures there is a separate tab for each project language in which the same setting options are available again.
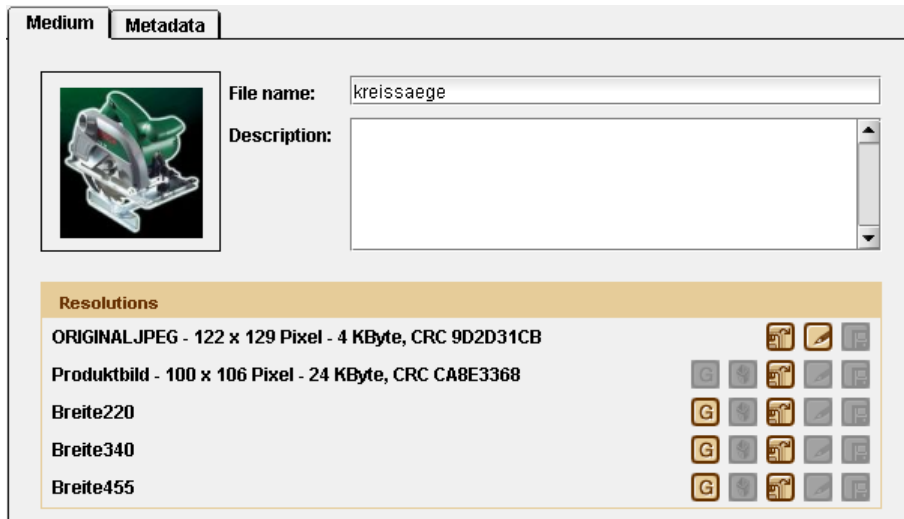
Figure 6-14: Medium View – Picture

A preview (thumbnail) of the current picture is automatically displayed in the top left-hand corner. If the Preview icon is clicked the program first tries to start the picture in the specified editor – if this action fails the default editor set in the operating system is started.

File Name: This row automatically contains the name without the corresponding file extension under which the selected medium is stored on the server. This name is used to link the picture on the website.

Description: This field can be used to enter an explanatory comment on this picture which, among other things, can be used as a tool tip on the website. The explanatory comment on the picture should of course be written in the relevant language.

The **Resolutions** area contains the options for changing the current picture. The top row describes the original pictures, as can be seen in the thumbnail. In addition, there is information on the picture type (here: JPG file), the picture size (here: 716 * 800 dots) and the file size (here: 108 kbyte). In addition, in the FirstSpirit JavaClient it is possible to have any number of different resolutions of a picture automatically rendered.

 Pre-Render Resolutions; click this icon to have a predefined resolution created for the picture. This is then stored on the server as a new picture and is displayed in the Edit screen form under Resolutions.

> ! *It is not advisable to render a resolution using this icon as the resolutions are automatically rendered by the system if and when necessary. This icon should only be used if the rendered resolution is to be subsequently edited. To do this it is necessary to save it as a separate picture.*

> ! *Resolutions rendered using this icon are not adjusted when the original picture is replaced as the system assumes manually changed pictures. In the case of automatically rendered resolutions on the other hand the old contents are discarded if the original picture is replaced and the resolutions are re-rendered and displayed again with the new original picture!*

Remove Pre-Rendered Picture; this icon is available for removing a resolution.

Select New Picture; click this icon to open a window in which you can move through the file structure at your workstation to search for the required picture.

Edit; click this icon to open the current picture in the entered editor. After the changes have been made in the external editor the picture is saved (it is not necessary to give file names or directory paths).

Save Local Copy; after the external editor has been closed you only have to click this icon. This integrates the edited picture from the local storage location in the Media Store of the FirstSpirit JavaClient.

6.6.2 Resolutions of images Edit from V4.1

> ! *This function is released for FirstSpirit Version 4.1 and higher only. Screenshots are therefore displayed in the new "LightGray" Look & Feel. The display can differ slightly in the "Classic" Look & Feel.*

From Version 4.1, FirstSpirit provides the option of making simple changes to pictures from the Media Store without having to use external image editing software. In this way, pictures can be, e.g. resized, rotated or mirrored.

The **Resolutions** area therefore differs from the description in Chapter 6.6.1 from
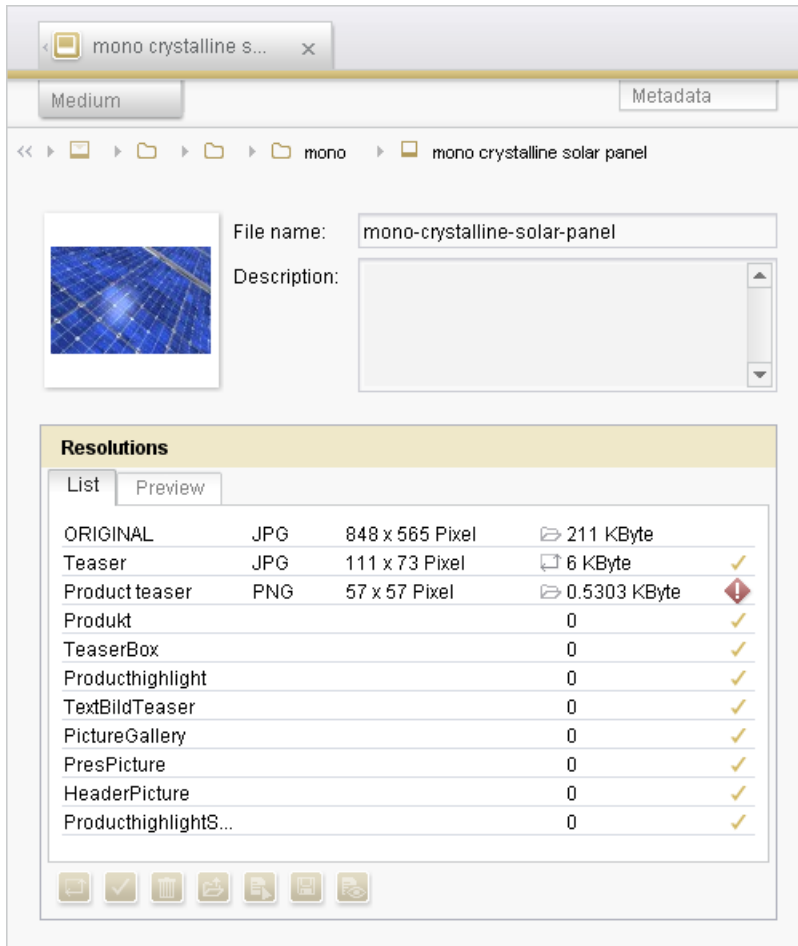
page 176.

6.6.2.1   List tab

Figure 6-15: Picture – Medium, List tab view

The resolutions defined in the project properties (cf. *FirstSpirit Manual for Administrators*) are listed on the List tab. From FirstSpirit Version 4.2, resolutions can also be displayed with a language-dependent display name.

The first line describes the original picture, as can be seen in the thumbnail. In addition, the picture type (here: JPG file), the picture size (here: 848 x 565 Pixel) and the file size (here: 211 Kbyte) are displayed.

The other resolutions are calculated automatically by the system when needed. It is also possible to upload their own images at any resolution or image sections to determine. Each uploaded image or edited is also displayed with picture type, picture size and file size.

Functions available for every resolution

The following functions can be performed for every resolution. To do this, the required resolution must be selected first by clicking the respective row. The icons are then only active if the corresponding function can be performed for the resolution.

Edit picture cutout; click this icon to open a window in which the tree structure of the Media Store is displayed.

> ! *The original picture can also be edited. In this case a confirmation prompt is displayed: "Do you really want to crop the original image?"*

Resolution checked: this icon is only active if the original picture of a cropped picture has been subsequently replaced. The resolution of the cropped picture is not automatically adjusted to the size of the new original picture and must therefore be manually checked and if necessary adjusted. Click this icon to change the ⬥ icon in the list of resolutions to ✓.

Delete stored image, this icon can be used to remove the editing of a resolution (by cropping, uploading a new picture or editing in the external editor); the resolution is rendered once again from the original picture. The original picture itself cannot be deleted with this icon.

Select new picture; click this icon to open a window in which you can move through the file structure at your workstation to search for the required picture.

> ! *If the size of the uploaded picture does not match the size of the specified resolution, the ⬥ icon is displayed in the list. The picture is then output on the website exactly the same as it was uploaded.*

Edit, click this icon to open the current picture in the editor entered for the respective file format in the *Global Settings / User Settings* or set in the operating system. If a resolution already exists, it is loaded in the editor; otherwise the original picture is opened. After the changes have been made in the external editor, the picture must be saved in the external editor (it is not necessary to give file names or directory paths). When the external editor is opened, a window is opened at the

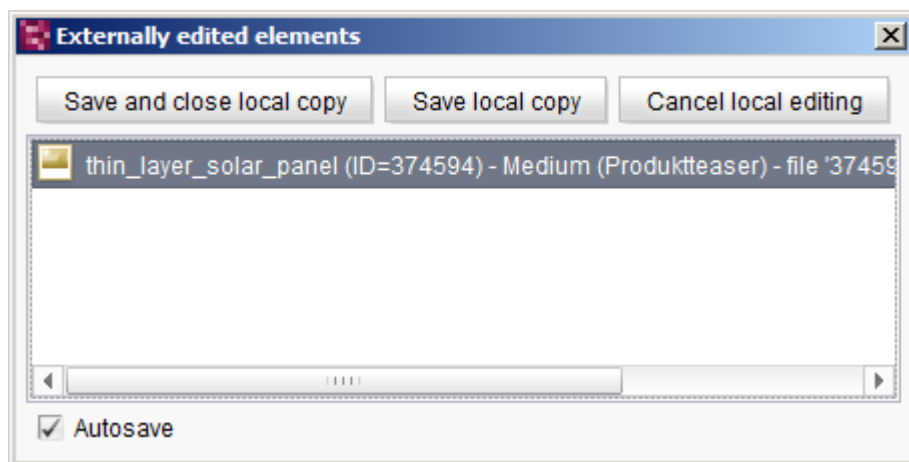same time in which all external opened elements are listed.

Figure 6-16: Externally edited elements

If the "Autosave" option is active, the changes saved in the external editor are always automatically adopted from the local storage location in the Media Store of FirstSpirit Client. (Only changes saved in the external editor can be automatically adopted.) When the window is closed, all open files can be discarded.

If the "Autosave" option is not active, then the changes saved in the external editor then have to be adopted in the Media Store of FirstSpirit Client by clicking the "Save and close local copy" or "Save local copy" button. If the "Cancel local editing" button is used the changes already saved externally are discarded and are not adopted in FirstSpirit Client.

If the size of the edited picture does not match the size of the specified resolution, the ◆ icon is displayed in the list. The picture is then output on the website exactly as it was saved after editing.

 Save local copy, if changes saved in the external editor are not automatically integrated in FiirstSpirit Client, click this icon to copy the edited picture from the local storage location into the Media Store of FirstSpirit JavaClient.

 Preview, opens a preview of the resolution in a pop-up window with 1:1 display, in the same way as it is also output on the website. Alternatively, this preview is also obtained by double-clicking the respective row with the resolution concerned.

In addition, the following icons are used in the list:

 Resized: this icon indicates that the picture has been edited using the dialog in Figure 6-17.

 Uploaded: this icon indicates that the picture has been uploaded using the icon or edited using the icon.

 Checked: this icon indicates that the deposited picture corresponds to the predefined resolution or can be automatically rendered by the system.

 Wrong resolution: this icon indicates that in the case of pictures uploaded using the icon or edited using the icon, the deposited picture does not correspond to the predefined resolution. The picture is output on the website precisely the same as it was deposited.

 Not yet checked: this icon is displayed if a resized resolution exists and if the original picture was replaced after the resizing. The resized resolution should be checked and if necessary adjusted. Several options are available for this:

- : the picture is edited internally: In this case the original picture is used. After saving, the icon in the list of resolutions becomes a ✓.

- : a new picture is uploaded. If the dimensions of the uploaded picture match the resolution, the icon in the list of resolutions becomes a ✓. Otherwise a is displayed.

- : the picture is edited in an external editor. The resolution of the edited picture is not checked and has to be manually confirmed using the icon. The icon in the list of resolutions becomes a ✓.

▪ ☑: the current sizing is retained without changes. The ❓ icon in the list of resolutions becomes a ✓.

> ❗ *Resolutions which were not edited beforehand are automatically re-rendered with the new original picture and displayed if the original picture has been replaced.*

6.6.2.2 Resizing function (from V4.1)

If the resizing function is called using the 🔲 icon, the system first checks whether a resolution already exists and whether it is based on the original picture or another picture:

▪ If no resolution exists yet, the original picture is opened for editing in the following dialog (see Figure 6-17).

▪ If a resolution exists which is based on the original picture, the original picture is also opened for editing in the following dialog (see Figure 6-17).

▪ If a resolution exists which is not based on the original picture but on an uploaded picture, a message appears: "This resolution is not based on the original medium. further resizing can cause loss in quality.", and then the uploaded picture is opened for editing in the following dialog (see Figure 6-17).

> ❗ *In this case, only the resized picture is saved, the sizing can only be made smaller at a later date.*

Figure 6-17: Resizing function for pictures

The selected image can now be edited in this dialog. Apart from the picture name, the current picture size is also displayed (here: 848 x 565 px).

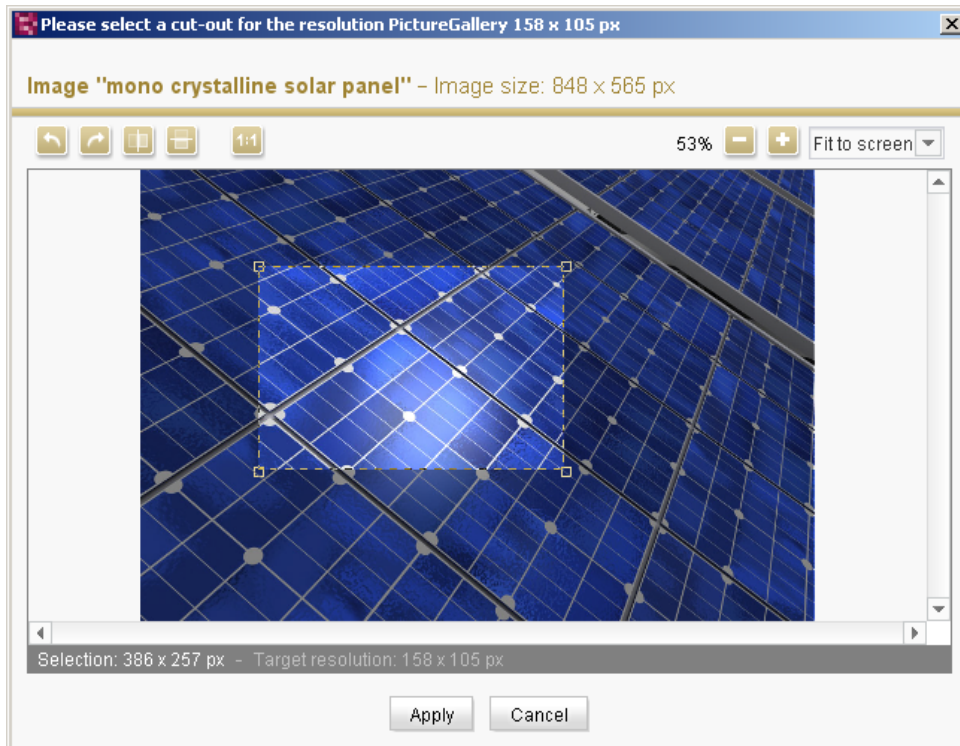The following editing options are available:

If the picture is larger than the target resolution, the frame can be used to select a suitable cut-out from the picture. The first time the picture is edited, as a default the frame is displayed centred with the size of the target resolution. The picture area outside the frame has a grey background. If the picture has already been edited, the frame is displayed with the size and in the position in which it was previously saved. Click inside the frame and drag to move it within the picture. The size of the frame can be changed with the help of the handles. In the case of resolutions for which the "Keep aspect ratio" option was activated in the project properties (cf. *FirstSpirit Manual for Administrators*) or which have a fixed size (e.g. 400 x 300 px), the frame can only be enlarged or reduced in size (sized up/down) proportionally. The frame therefore only has handles in the corners. In the case of resolutions without a specific side ratio, the frame also has handles on the sides. The corner handles can be used to proportionally enlarge or reduce the size of the picture by simultaneously pressing the Shift key.

> **!** *If the original picture is to be edited, a frame must be manually pulled onto it first.*

Selection: These pixel dimensions give the current size of the frame. If a smaller cut-out is selected than the target resolution, the size details appear in yellow lettering. In this case, a loss in quality can occur as system will automatically scale the picture larger. If a larger cut-out is selected than the target resolution, the system automatically scales down the picture to the resolution size.

Target resolution: These pixel dimensions give the size defined for this resolution in the product properties (cf. *FirstSpirit Manual for Administrators*).

Rotate 90° to the left: this icon can be used to rotate the picture through 90° to the left. If necessary, the selection frame must be moved or reduced in size within the picture so that the selection can be accepted.

Rotate 90° to the right: this icon can be used to rotate the icon through 90° to the right. If necessary, the selection frame must be moved or reduced in size within the picture so that the selection can be accepted.

Flip horizontally: this icon can be used to mirror the picture horizontally.

Flip vertically: this icon can be used to mirror the picture vertically.

Selection in scale 1:1: this icon can be used at any time to reset the selection frame back to the size of the target resolution.

> **!** *If the picture is smaller than the target resolution, this icon is inactive, as well as in the original picture.*

Reduce zoom size: as a default, the size of the picture to be edited is adjusted to the size of the edit window. This icon can be used to reduce the size of the zoom to enable a larger area of the picture to be viewed.

Increase zoom: this icon can be used to increase the zoom to examine a cut-out

of the picture in greater detail. A pixel grid is also provided for high zoom levels.

Alternatively, the mouse wheel can be used to zoom into and out of the picture.

Select ⌄ , this combobox can be used to select predefined zoom levels. Use the "Fit to screen" to return to the initial zoom view.

The Apply button is used to save the picture with its current edit status and the pop-up window closes. If the edited picture is based on the original picture, if necessary, another picture cut-out can be selected later. If the picture is not based on the original picture, the resized picture is saved, the sizing can only be made smaller at a later date. The same applies if the original picture has been edited.

The picture is now displayed in the list of resolutions with its file type, picture size and file size. The ⊡ icon indicates that the picture has been edited using the resizing function.

6.6.2.3   Preview tab
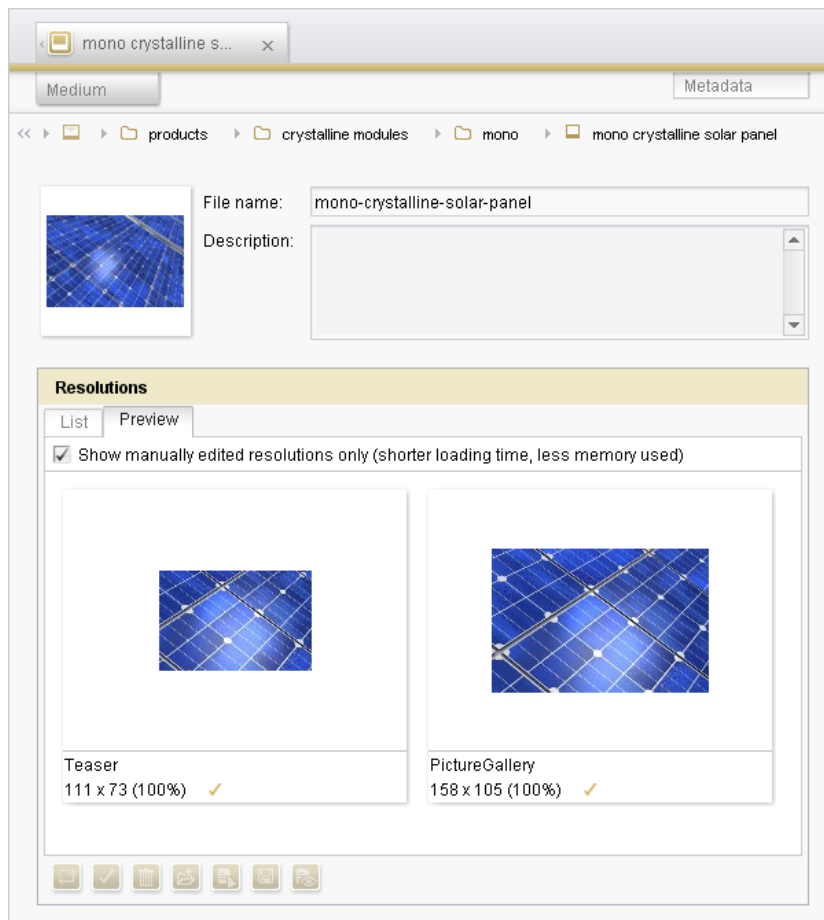
Figure 6-18: Picture – Medium, Preview tab view

A preview of each manually edited resolution is displayed on the Preview tab, e.g. to enable the display to be more easily checked. If the "Display manually edited resolutions only" checkbox is disabled, all resolutions are displayed (i.e. the edited resolutions **and** the automatically generated ones).

Below the resolution name (in Figure 6-18 for example "Teaser"), the dimensions of the resolution and the respective status (✓, ⬥ or ❓) are given as in the list view. A percentage smaller than 100 indicates that the preview picture is displayed with reduced size. Double-click the preview picture to display it in a pop-up window with 1:1 display.

The icons below the picture displays can be used to perform the same functions as in the list view (see Chapter 6.6.2.1 page 179).

6.6.3   Files

Files means all other media formats apart from pictures. These can be, e.g. PDF, MP3 or even video files.

The setting options for language-dependent and language-independent files are identical. For language-dependent files there is a separate tab for each project language in which the same setting options are available again.
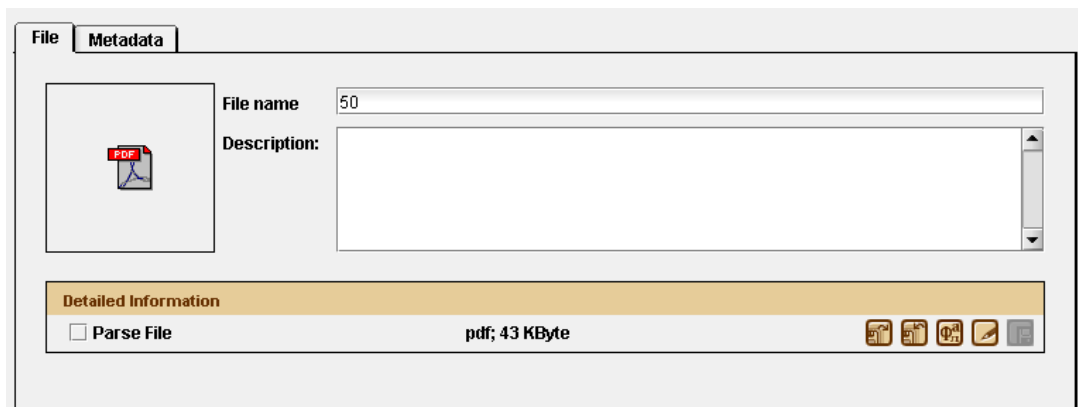

Figure 6-19: Medium View – File

The file format is shown in the top left-hand corner. Click the symbol to display the file in the corresponding editor (e.g. Acrobat Reader for a PDF document). Press the SHIFT key at the same time as clicking the file to view it in the editor entered in the user settings. If the SHIFT key is pressed on clicking the file and there is no editor entered in the settings the file opens in a supplied inline editor.

File name: This row automatically contains the name without the corresponding file extension under which the selected medium is stored on the server. This name is used to link the file on the website.

Description: This field can be used to enter an explanatory comment on this file which, among other things, can be used as a tool tip on the website. The explanatory comment on the file should of course be written in the relevant language.

The Detailed Information area contains the options for changing the current file. The middle contains the file extension details (here: pdf) and the file size (here: 3 kbyte). In addition the following options are available for making changes to the file:

Parse File: If this option is deactivated, media which contain variables still to be resolved are simply run through. If this option is activated the code of the medium is checked and all variables are resolved.

Select New File; click this icon to exchange the current medium. A window opens in which you can move through the file structure on your workstation to click the required new medium. In the case of language-dependent files a separate file must be selected for each language.

Generate Local Working Copy; this icon can be used to save the file on the local computer too.

Change Medium Encoding; if the medium is a text file it can be useful to change the encoding to adjust the file to certain county-specific notations or styles. Click this icon to open a new window for selecting the required encoding.

Edit; you can of course also edit a file. After entering an editor in the user settings in the Global Store, click this icon to open the entered editor with the current file. After the changes have been made in the external editor you can save the medium there with "Save" without having to worry about the file name or directory path.

Save Local Copy; after the external editor has been closed, in FirstSpirit™ you only have to click this icon. This integrates the edited medium from the local storage location in the Media Store of the FirstSpirit™ JavaClient.

Compare with Old Version; this icon is available to compare a file with an older version after editing. Here you can select an older version (if available) for comparison and view the results using the editor defined in the user settings.

6.7 Processing of media in the integrated preview (from V4.2R4)

If using applications in the Integrated preview, please note that FirstSpirit provides the interfaces required for the application integration, but in general does not have any influence on the integrated applications themselves. Integrated external applications are not part of the FirstSpirit product. Among other things, this means that responsibility for the function of the integrated applications lies with the manufacturer of the application or with the customer or partner who implements the application. (See also FirstSpirit Release Notes Version 4.2R4, chapter 3 "The FirstSpirit AppCenter").

In order to edit pictures in the Integrated preview, the picture to be edited must be set in Edit Mode (locked) in the Media Store and the required resolution selected

with the click of the mouse.

The different resolutions are displayed in the ribbon of the integrated preview. If more resolutions exist than can be displayed in the current height, it is possible to scroll up or down in the ribbon. The original resolution and manually edited resolutions are displayed with strong colours, which are automatically rendered by the system; resolutions which have not been edited are displayed in toned down (pale).
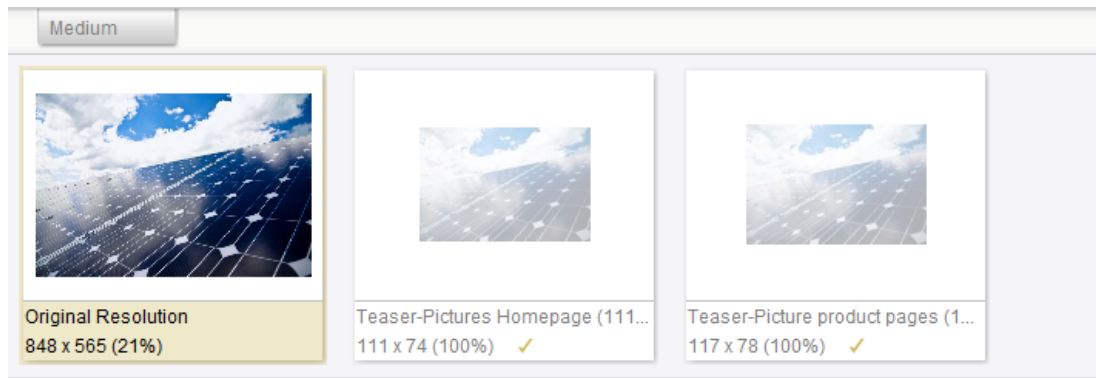

Figure 6-20: Resolutions in the Integrated preview

Different editing options are available, depending on the settings made in the "View" menu under the "Integrated preview" and "Graphic engine" points. These options always relate to the resolution selected in the ribbon (yellow border). Below it is the area in which the picture can be edited using the respective available functions.

Initially, the cropping function is also available in the integrated preview if the "Edit image cutout" icon is clicked (see chapter 6.6.2 page 178).

Changes to pictures/resolutions first have to be copied from the local storage location into the Media Store of FirstSpirit by clicking the icon. Finally, the changes must be saved using the "Save" or "Exit Editing" button.

Click the icon again to exit editing; any changes not saved are not adopted in FirstSpirit.

6.7.1 Java Image Editor

If the "Graphic engine – Java Image Editor" option is activated, pictures and individual resolutions can also be edited using Java Image Editor via the **Enhanced**

image editing button.

Note: The release for use of the Java Image Editor is explicitly given "without function guarantee" for the application itself. I.e. e-Spirit does not provide a guarantee for the image processing functions, neither explicitly nor implicitly, and instead they are released for use "as is". If the use of image processing functions is a critical function for production, external image processing software should be used with the corresponding manufacturer support. Java Image Editor is currently available under MacOS only to a very limited extent.

After clicking this button, the editor with its editing functions and tools and the image file with the resolution selected in the ribbon are displayed in the editing window below the ribbon:
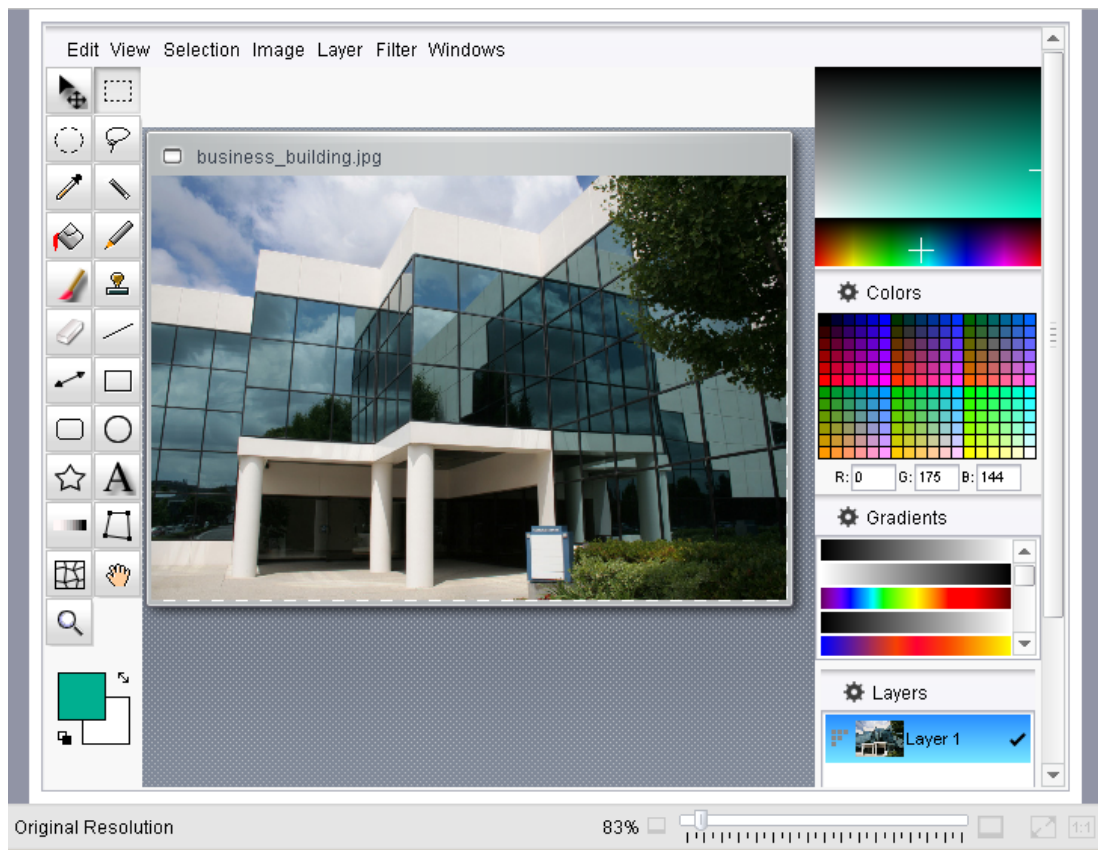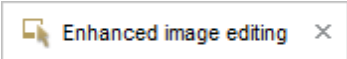
Figure 6-21: Enhanced image editing – Java Image Editor

Changes to pictures/resolutions first have to be copied into the Media Store of FirstSpirit by clicking the [icon] icon. You can then continue editing the picture. Finally, the changes must be saved using the "Save" or "Exit Editing" button.

Click the [Enhanced image editing ✕] button again to close the editor; any changes not saved are not adopted in FirstSpirit.

The functions available in Java Image Editor are comparable with those in other relevant image editing/processing programs.

> **!** *Several menus, tooltips and configuration dialogs (e.g. for filters) cannot be localised for technical reasons and are therefore only available in English.*

> **!** *Several of the keyboard shortcuts displayed in the menus do not work for the editor, but instead perform the function known to date in JavaClient.*

6.7.1.1 Toolbar

Some tools provide enhanced configuration options. These are displayed above the image, below the menu bar.

Selection tools

This icon can be used to move a selection or to resize it using the selection grab handles.

Use these icons to select a square / rectangular, round / oval or lasso for user-defined shape of the image for further editing. Such a selection can, e.g. be applied using the "Edit" menu, modified u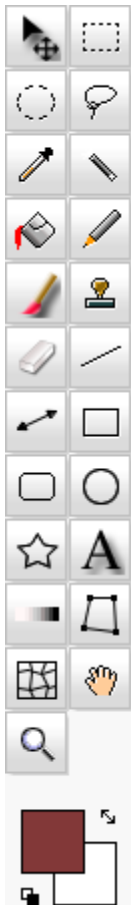sing the "Selection" menu and moved using the icon. All painting tools (see below) are only applied within a selection. Additional functions are available with the Ctrl key pressed.

This icon can be used to select contiguous areas, which have the same or similar colour.

Use this icon to select through the colour of pixels in the image.

Painting and drawing tools

Use this icon to fill adjacent areas with similar colour.

This icon can be used to draw a hard-edged freehand line.

This icon is used to draw softer coloured lines.

Use this icon to duplicate a defined area of the image.

Use this icon to make pixels of the image transparent.

These icons can be used to create straight lines and arrows with defined width and fill method.

These icons can be used to create filled or unfilled squares /

rectangles, with angular or rounded corners, circles / ovals or stars.

View

If the image window has scroll bars due to the zoom level set, this icon can be used to move the image within the window.

Use this icon to increase / reduce (Ctrl) the zoom level (zoom in/out).

Other

Use this icon to insert a layer with text into the image.

This icon is used to add colour gradients.

These icons are used to modify the perspective of the image by moving the corner points or to distort the image using a grid.
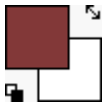
These icons are used to display the current foreground and background colour. They can be swapped using the arrow icon. Use the black&white icon to select black and white as the background/foreground colour.

6.7.1.2   Menu Area

"Edit" menu

This menu is used to apply selections: They can be

- cut (**Cut**)
- copied (**Copy**)
- inserted (**Paste**)
- deleted (**Clear**)

- If several layers exist on top of each other, **Copy Merged** is used in the selected area to create a copy of all visible layers reduced to one layer.
- In addition, editing steps can also be undone (**Undo**).

"View" menu

This menu is used to control the view of the image:

- Use the **Zoom In** and **Zoom Out** entries to gradually zoom the image.
- Use the **1:1**, **2:1**, **4:1**, **8:1** and **16:1** entries to set a specific zoom level.
- The **Show Grid** entry display a grid overlay on the image.
- Use the **Show Selection as Mask** entry to display the selection as a mask.

"Selection" menu

Use this menu to modify an existing selection:

- **Select All:** Selects the whole image.
- **Select None:** Empties the selection so that there is no longer any selection in the image.
- **Invert:** Selects the area of the image previously not selected.
- **Feather…:** Creates a selection with soft edges. The radius can be specified.
- **Grow:** Increases the selection by the specified radius.
- **From Layer Transparency:** Selects all transparent areas of the image.
- **New Layer from Selection:** Creates a layer on the basis of the selection.

"Image" menu

Use this menu to modify the whole image:

- **Crop:** Removes all parts of the image except for the selection.
- **Image Size…:** This can be used to change the size of the image.
- **Flip Horizontal / Vertical / Diagonal:** Mirrors the image horizontally / vertically along the vertical axis or diagonally about the transverse axis.
- **Rotate 90 / -90 / 180 / …:** Rotates the image through 90° in the clockwise or anticlockwise direction, or through 180° in a clockwise direction or by the number of degrees defined by the user.
- **Fill Selection:** Fills the whole current selection with the selected foreground colour.

"Layer" menu

This menu is used to create, modify and delete layers. On exiting the Java Image Editor, several layers are merged to form one; the next time the Editor is started, only this level is available.

- **New Layer / via Copy / via Cut:** Creates a new empty layer, a layer with the content of the current selection or a layer with the current selection and removes the selection from the existing layer.
- **Delete / Duplicate Layer:** Deletes or duplicates the currently selected layer.
- **Rename…:** This menu entry can be used to change the name of the layer. The new name is not saved on exiting the Editor and is no longer available the next time the Java Image Editor is started.
- **Move Up / Down:** Moves the current layer one position up / down.
- **Merge Down:** Merges the currently selected layer with the layer(s) underneath it.
- **Flip Horizontal / Vertical / Diagonal:** Mirrors the selected layer horizontally / vertically along the vertical axis or diagonally about the transverse axis.
- **Rotate 90 / -90 / 180:** Rotates the selected layer through 90° in a clockwise or anticlockwise direction, or through 180° in a clockwise direction.

"Filter" menu

This menu can be used to apply filters to the image, the current layer or the current selection.

- **Repeat Last Filter:** Applies the last applied filter again.
- **Show Last Filter:** Displays the last applied filter or the corresponding configuration dialog.

"Windows" menu

This menu can be used to hide or show the palettes on the right-hand side.

6.7.1.3 Palettes

- **Colours:** The foreground colour can be selected here with a click.
- **RGB colours:** The foreground colour can be selected here by setting RGB values.
- **Swatches ("Colors"):** Here the foreground colour can be selected from a library. As a default, a palette of colours is used, which can be used, platform-independent, by all browsers ("web colours"). If necessary, you can import your own libraries (and save them again later). Uploaded colour libraries are not saved on exiting the Editor and are no longer available the next time the Java Image Editor is started.
- **Gradients:** A colour gradient can be selected here. The existing ones can be used, edited or the user's own gradients can be loaded. Uploaded or edited colour gradients are not saved on exiting the Editor and are no longer available

the next time the Java Image Editor is started.

▪ **Layers:** New layers can be created, edited or deleted here. See also Chapter 6.7.1.2 page 194. In addition, fill methods can be set for the current layer. On exiting the Java Image Editor, several layers are merged to form one; the next time the Editor is started, only this level is available.

6.7.2 Simple image editing (picnik)

If the "Graphic engine – Simple image processing (Picnik)" option is activated, then photos and individual resolutions can be edited using the Picnik online image processing service via the **Enhanced image editing** button.

Note: An active connection with the internet is required to use this option. The pictures to be edited are in fact uploaded onto the server of the respective provider and are edited there. This should be taken into account with regard to data protection issues if these applications are used.

After clicking the **Enhanced image editing** button, the editor and its editing functions and tools as well as the image file with the resolution selected in the ribbon are displayed in the editing window below the ribbon:
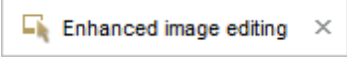

Figure 6-22: Simple image editing (picnik)

Changes to pictures/resolutions first have to be copied from the local storage

location into the Media Store of FirstSpirit by clicking the **Save in FirstSpirit** button. You can then continue editing the picture. Finally, the changes must be saved using the "Save" or "Exit Editing" button.

Click the **Enhanced image editing ✕** button again to close the editor; any changes not saved are not adopted in FirstSpirit.

6.7.3 Enhanced image editing (Pixlr)

If the "Graphic engine – Enhanced image processing (Pixlr)" option is activated, then photos and individual resolutions can be edited using the Pixlr online image processing service via the **Enhanced image editing** button.

Note: An active connection with the internet is required to use this option. The pictures to be edited are in fact uploaded onto the server of the respective provider and are edited there. This should be taken into account with regard to data protection issues if these applications are used.

After clicking the **Enhanced image editing** button, the editor and its editing functions and tools as well as the image file with the resolution selected in the ribbon are displayed in the editing window below the ribbon:
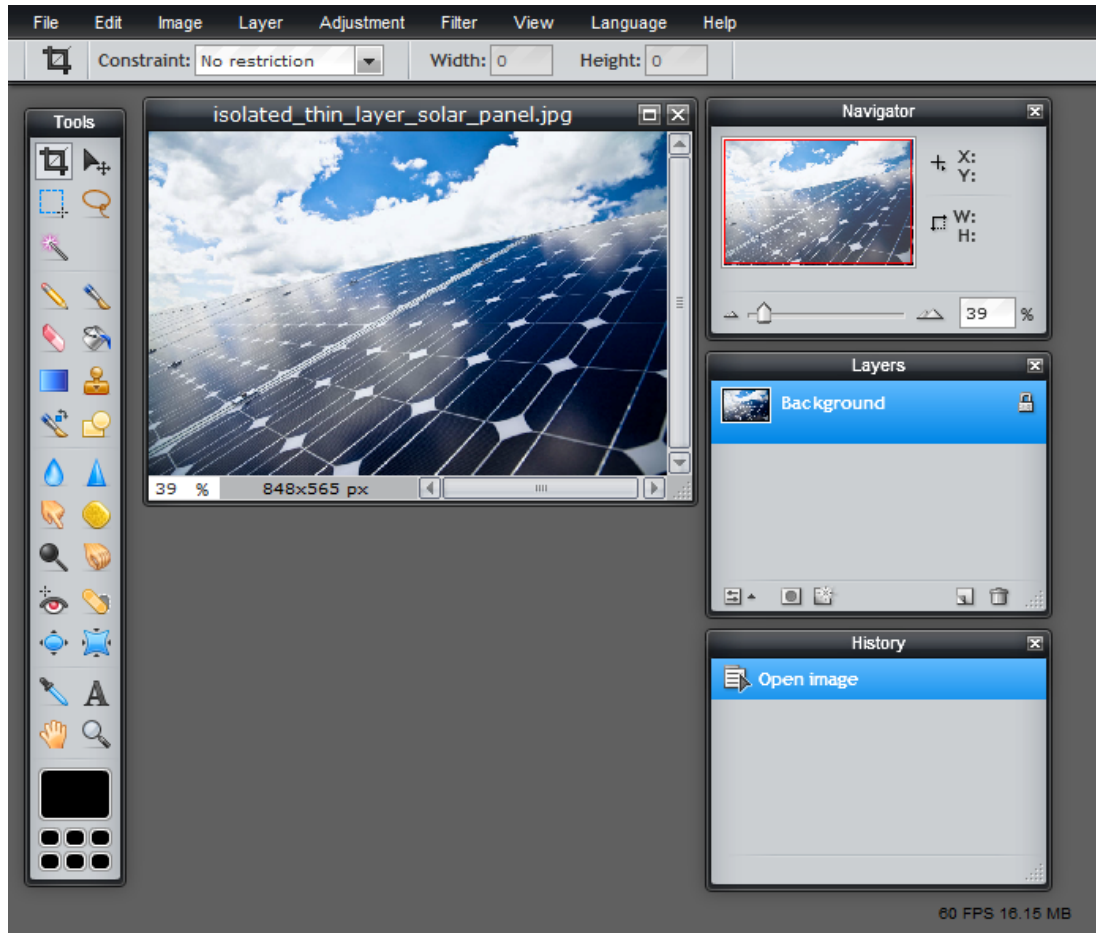
Figure 6-23: Enhanced image editing (pixlr)

Changes to pictures/resolutions first have to be copied from the local storage location into the Media Store of FirstSpirit by clicking the Menu item "Save…" in the Menu "File". You can then continue editing the picture. Finally, the changes must be saved using the "Save" or "Exit Editing" button.

Click the [Enhanced image editing ✕] button again to close the editor; any changes not saved are not adopted in FirstSpirit.

6.7.4 Microsoft Office (Windows only)

If the "Office engine – Microsoft Office (Windows only)" option is activated, then files can be edited directly in the integrated preview.

If you switch to Edit mode for the file to be edited, a suitable Microsoft Office application with its editing functions and tools automatically opens in the area of the
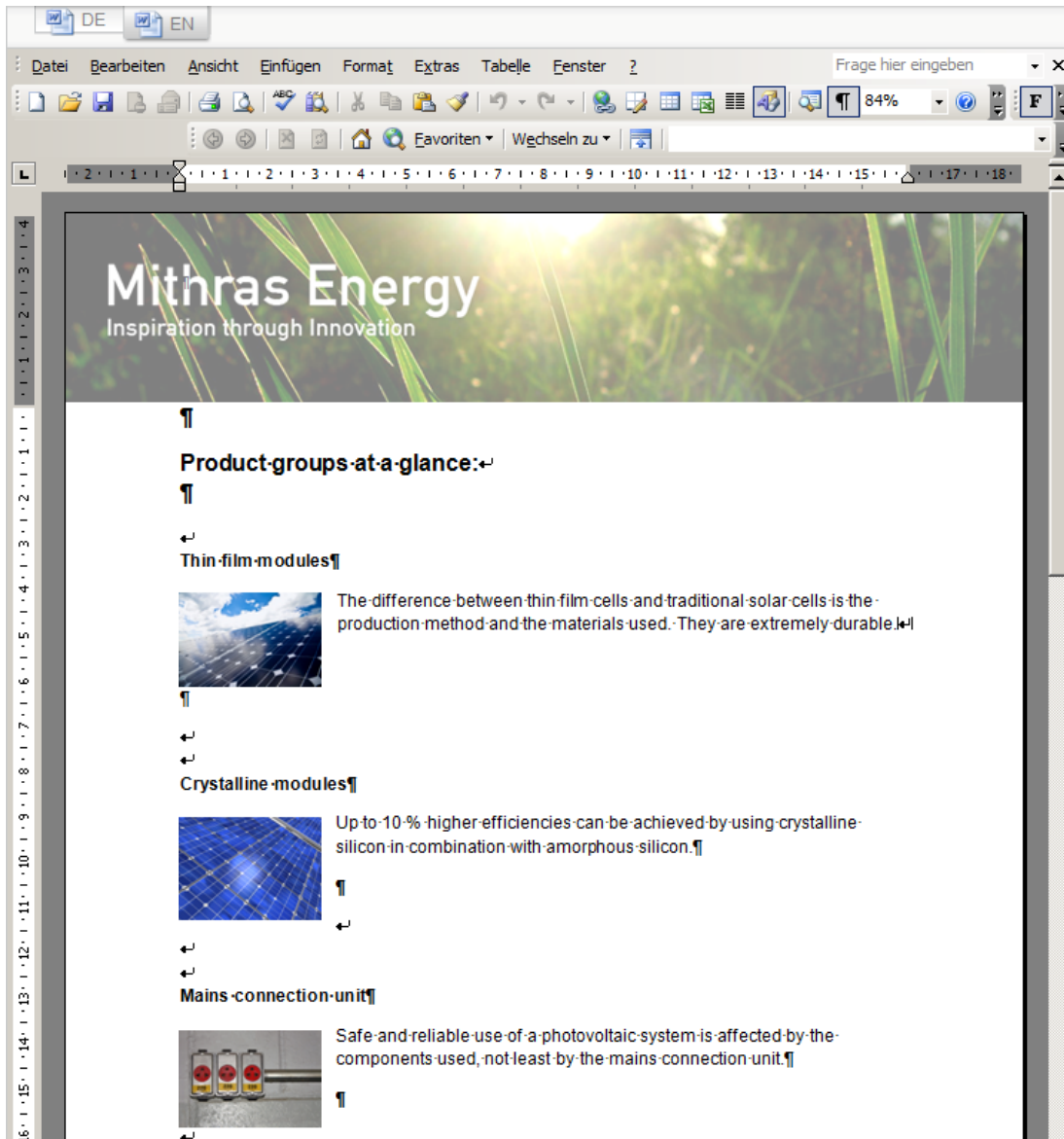
integrated preview.

Figure 6-24: File editing with Microsoft Office

Changes to a file do not have to be saved within the application, but instead, directly in FirstSpirit using the "Save" function or by changing to View mode.

6.7.5 OpenOffice (BETA, not MacOS)

If the "Office engine – OpenOffice (BETA, not MacOS)" option is activated, then files can be edited directly in the integrated preview.

If you switch to Edit mode for the file to be edited, a suitable OpenOffice application

with its editing functions and tools automatically opens in the area of the integrated preview.
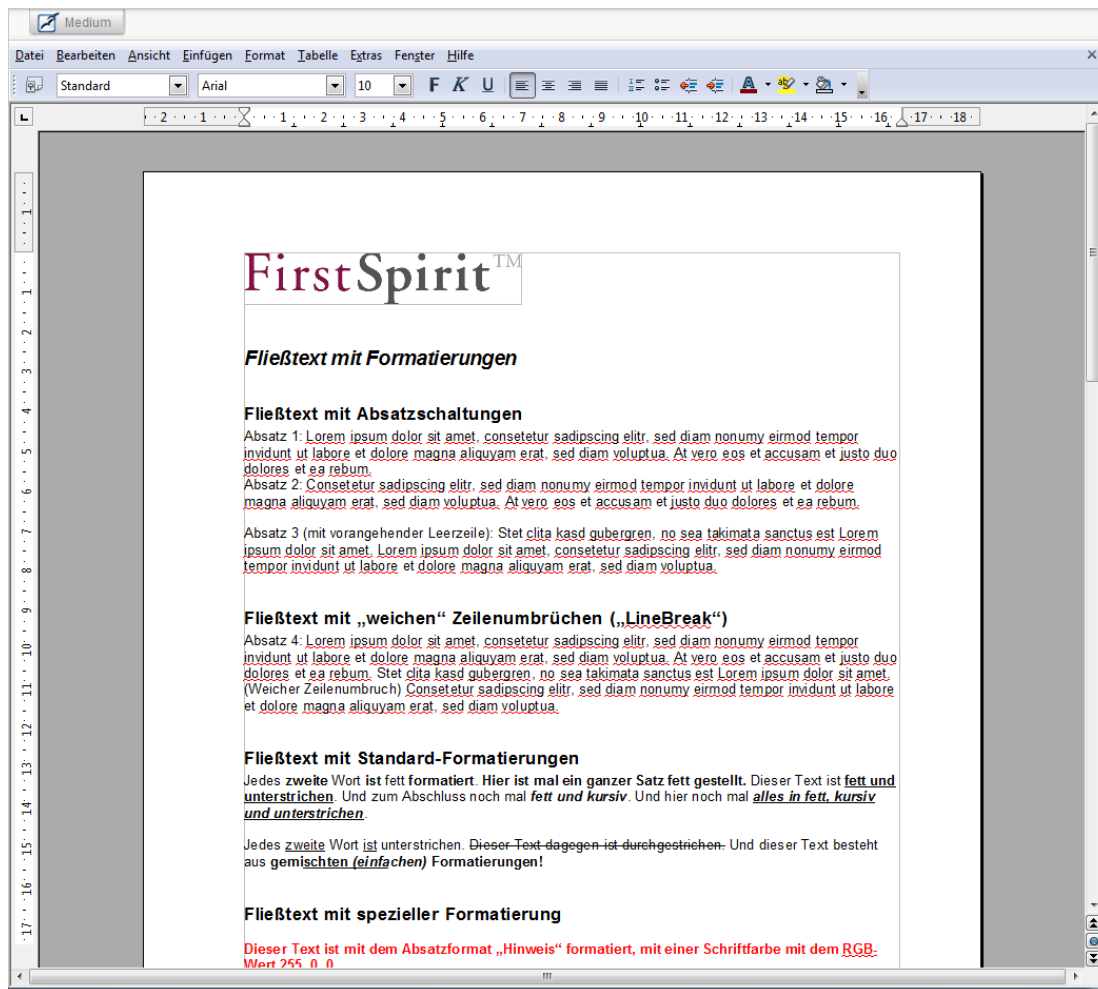

Figure 6-25: File editing with OpenOffice

Changes to a file do not have to be saved within the application, but instead, directly in FirstSpirit using the "Save" function or by changing to View mode.

6.8   Media Import Wizard

The Media Import Wizard helps to import large quantities of media if they, e.g. have been prepared using an external program. If these media exist in different languages and resolutions the wizard ensures that FirstSpirit imports media objects which contain these resolutions.

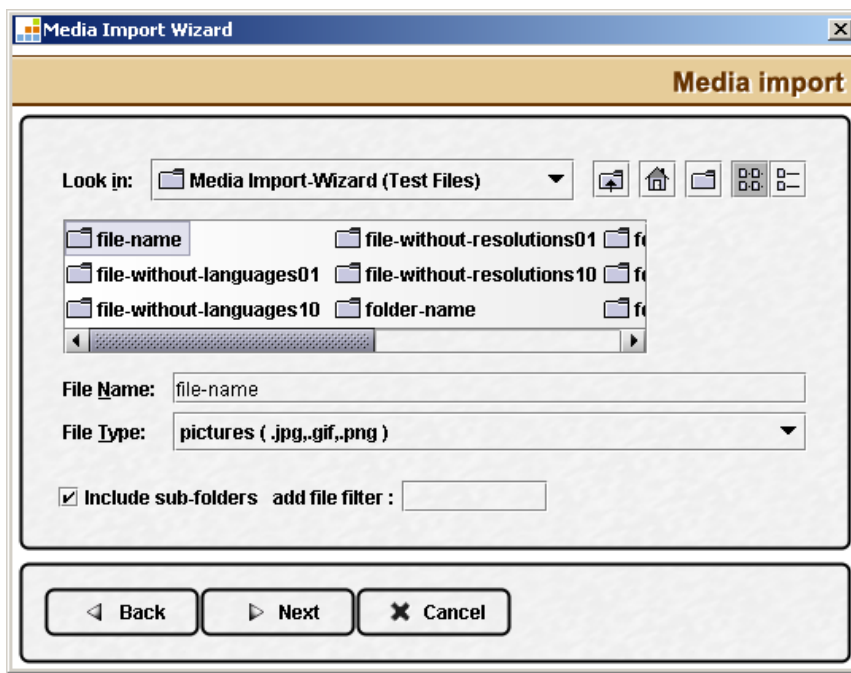### 6.8.1   Media Import Wizard – Step 1 of 9

Figure 6-26: Media Import Wizard – Step 1

In the first step you must define the folder in the computer's file system in which the media to be imported are located. The media do not have to be directly in this folder, they can also be located in sub-folders.

File Name: The name of the folder in which the media for the import are located.

This may not be opened with a double-click but must be selected from the next higher level with a single click!

File Type: Here you can choose between All File Types and Pictures (.jpg, .gif, .png)

According to the settings in the project configuration (see *FirstSpirit Manual for Administrators*), the selection of media can be restricted to specific file sizes and file types from FirstSpirit version 4.1. In this case media which are bigger than the maximum size which is defined in the project configuration and/or which have a file extension which is not allowed, are not imported by the Media Import Wizard (see Chapter 6.2.1.1 page 162).

Include Sub-Folders: If this option is activated all media in the lower-level folders are imported too.

Press the [▷ Next] button to proceed to the next step:
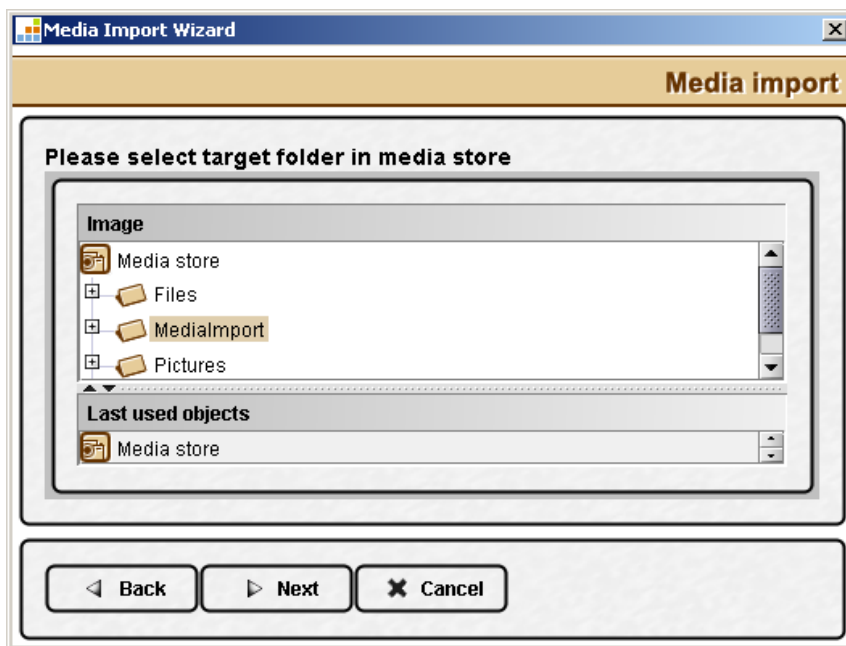
6.8.2 Media Import Wizard – Step 2 of 9

Figure 6-27: Media Import Wizard – Step 2

In the second step you must select the folder in the Media Store in which the imported media are to be integrated.

> ❗ *If Permission check is activated, folders for which the user does not at least have permissions to create objects and folders are denoted by green lettering. These folders can then not be selected.*
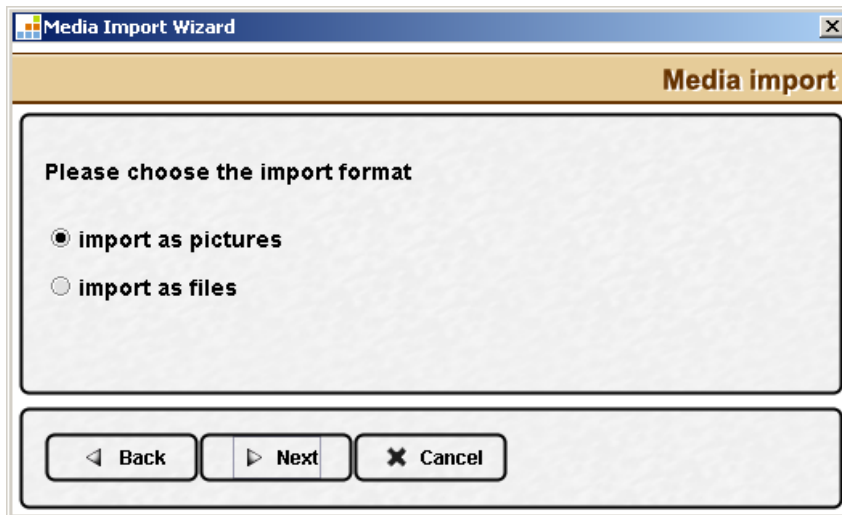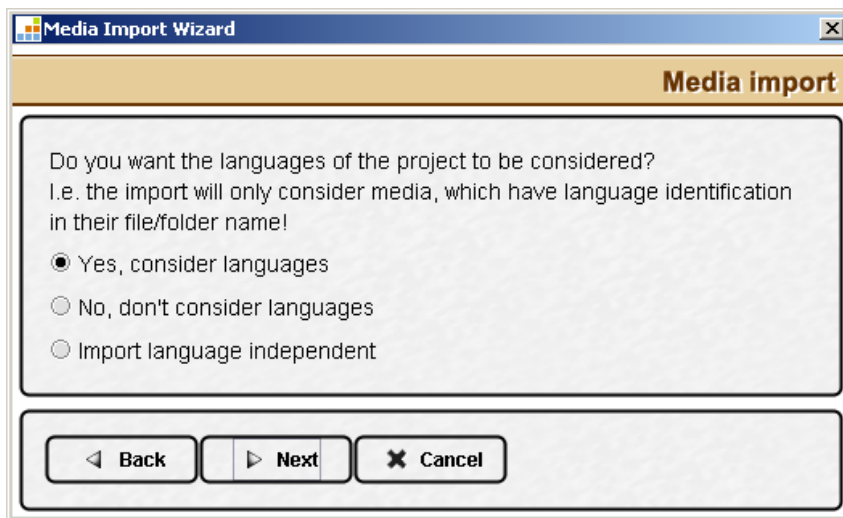
6.8.3 Media Import Wizard – Step 3 of 9

Figure 6-28: Media Import Wizard – Step 3

In the third step the user selects whether the imported media are to be created in Media Store as pictures or as files.

6.8.4   Media Import Wizard – Step 4 of 9

Figure 6-29: Media Import Wizard – Step 4

In the fourth step the user chooses the extent to which the project's languages are to be taken into account in the import.
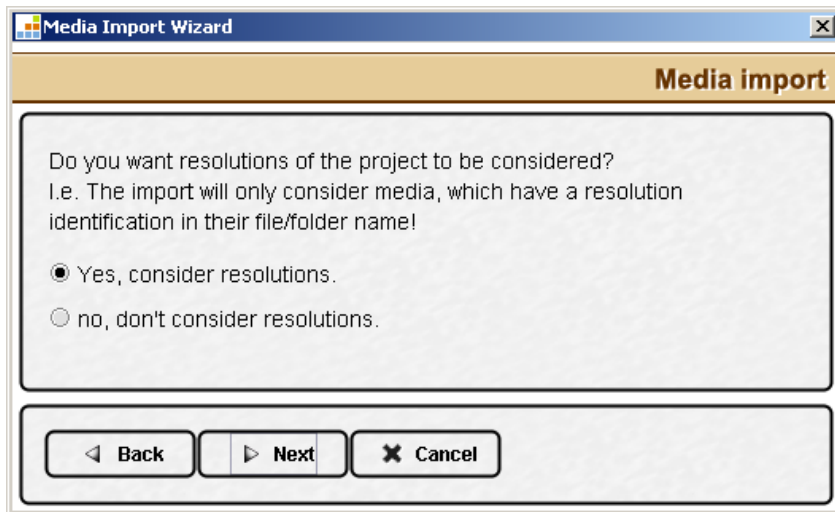
Yes, consider languages: If the project has several languages the media can also be directly assigned to the individual project languages during the import. This requires that the media names or folders in which the media are located have a label or flag to identify the corresponding language.

No, do not consider languages: This item should be selected if there are several languages in the project but the existing media are for one language only. A language-dependent medium is then created but a medium is imported for one project language only (usually the master language).

Import language-independent: This option must be selected if language-independent media are to be generated in the Media Store.

6.8.5 Media Import Wizard – Step 5 of 9

Figure 6-30: Media Import Wizard – Step 5

In the fifth step the user chooses whether the various resolutions of the project are to be taken into account in the import.

Yes, consider resolutions: If the project has several resolutions the media can also be directly assigned to the individual project resolutions during the import. This requires that the media names or folders in which the media are located have a label / flag to identify the corresponding resolution.

No, do not consider resolutions: This option is to be selected if media for one resolution only are to be imported (original resolution only).
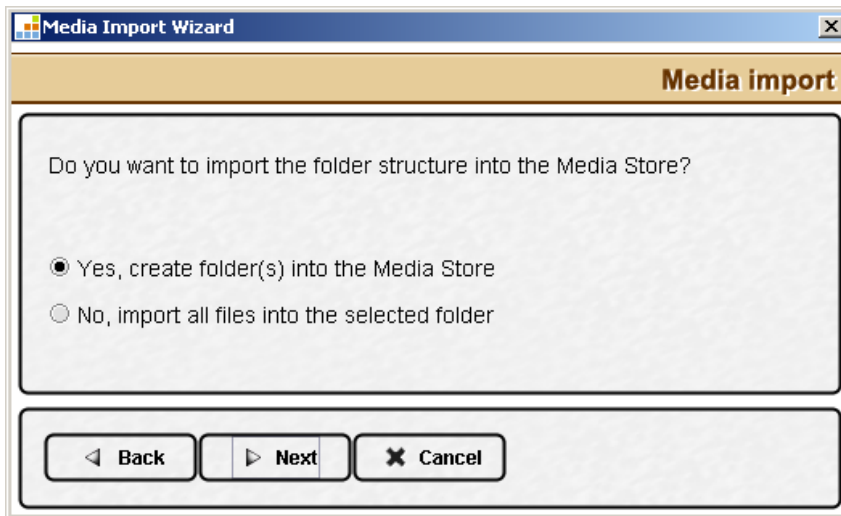
6.8.6 Media Import Wizard – Step 6 of 9

Figure 6-31: Media Import Wizard – Step 6

In the sixth step the user chooses whether the folder structures in the computer's file system are to be imported into the Media Store.

This step is omitted if the **Include Sub-Folders** option was not activated in the first step.

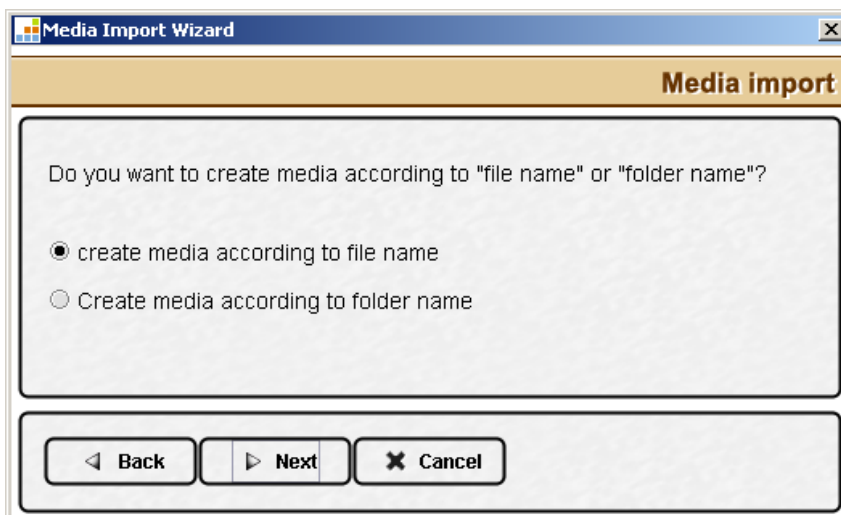### 6.8.7 Media Import Wizard – Step 7 of 9

Figure 6-32: Media Import Wizard – Step 7

In the seventh step the user makes a choice regarding naming of the media. This

selection depends on the structure of the data in the file system.

Create media according to file names: This option must be selected if the files to be imported have been named using a uniform convention. I.e. the different languages and resolutions are integrated in the file names.

Create media according to folder names: This option must be selected if the different languages and resolutions are characterised by the structure of the folders in which they are created.

6.8.8 Media Import Wizard – Step 8 of 9

The eighth step is highly dependent on the decisions made in the first seven steps. More precise details must then be given for the import according to the settings made to date.

Example 1:

Each medium has a language identifier and a resolution name integrated in the file name.
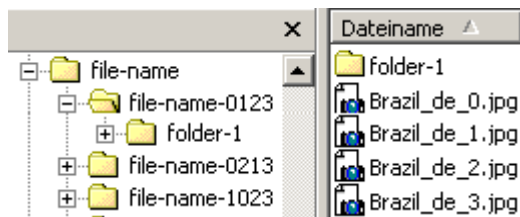

Figure 6-33: Create media via file names
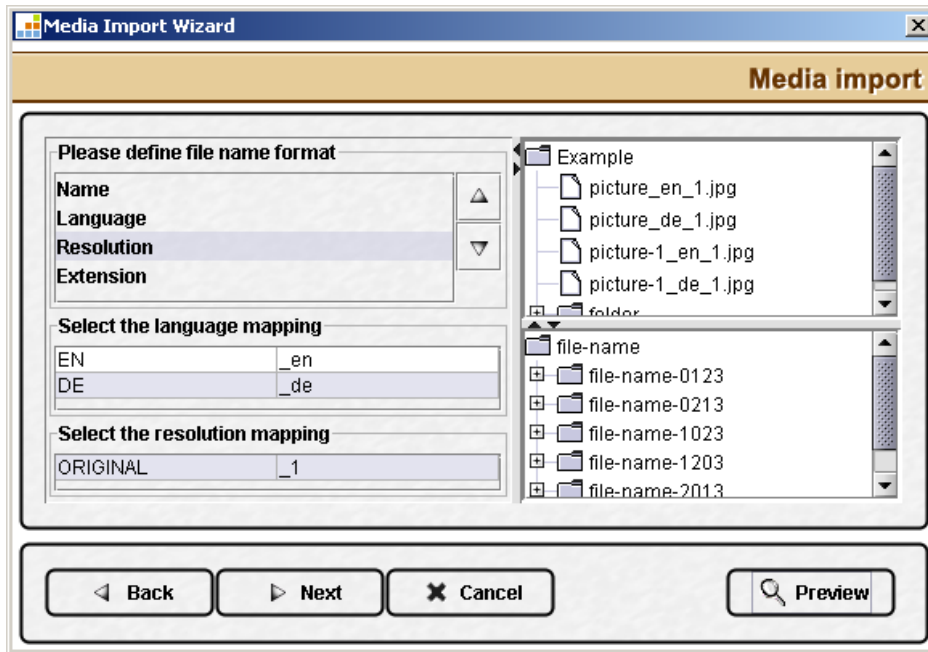
The eighth step is then as follows:

Figure 6-34: Media Import Wizard – Step 8 (Example 1)

File name format: The file names in the folder must be chosen so that they conform to the media import conditions. The order in which the information is integrated in the file name must be given here. The individual elements can be moved one position up or down using the arrow keys on the right.

Language mapping: All the project's languages are given here. It is necessary to enter the abbreviation used to identify each language in the file name. Click the respective language to select the corresponding abbreviation via a combo box or enter it manually.

Resolution mapping: All the project's resolutions are given here. It is necessary to enter the abbreviation used to identify each resolution in the file name. Click the respective resolution to select the corresponding abbreviation via a combo box or enter it manually.

The [Preview] button activates a window containing an overview of the objects to be imported into the Media Store.

<u>**Example 2:**</u>

The language identifier and name of the resolution are integrated in the folder structure. If the media to be imported are located in the following folder structure the **Create media according to folder names** option must be selected in the seventh step.
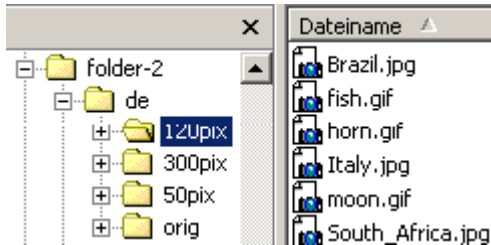

Figure 6-35: Create media via folder names

The eighth step is then as follows:
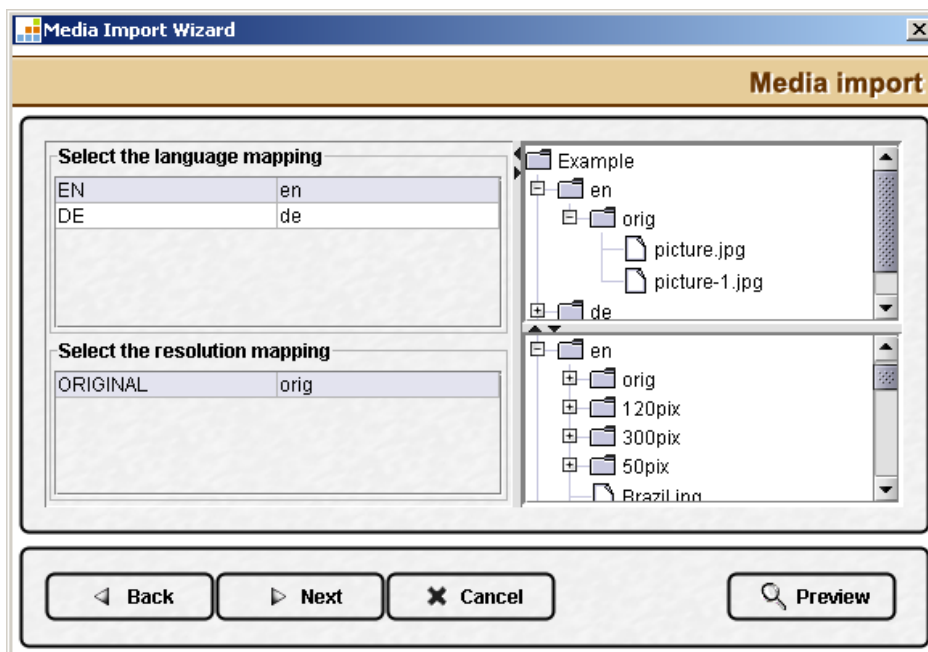
Figure 6-36: Media Import Wizard – Step 8 (Example 2)

Language mapping: All the project's languages are given here. The user must give the name of the folders in which the media for the corresponding language are filed for each language. Click the respective resolution to select the corresponding abbreviation via a combo box or enter it manually.

Resolution mapping: All the project's resolutions are given here. The user must

give the name of the folders in which the media for the corresponding resolution are filed for each resolution. Click the respective resolution to select the corresponding abbreviation via a combo box or enter it manually.

The [Q Preview] button activates a window containing an overview of the objects to be imported into the Media Store.

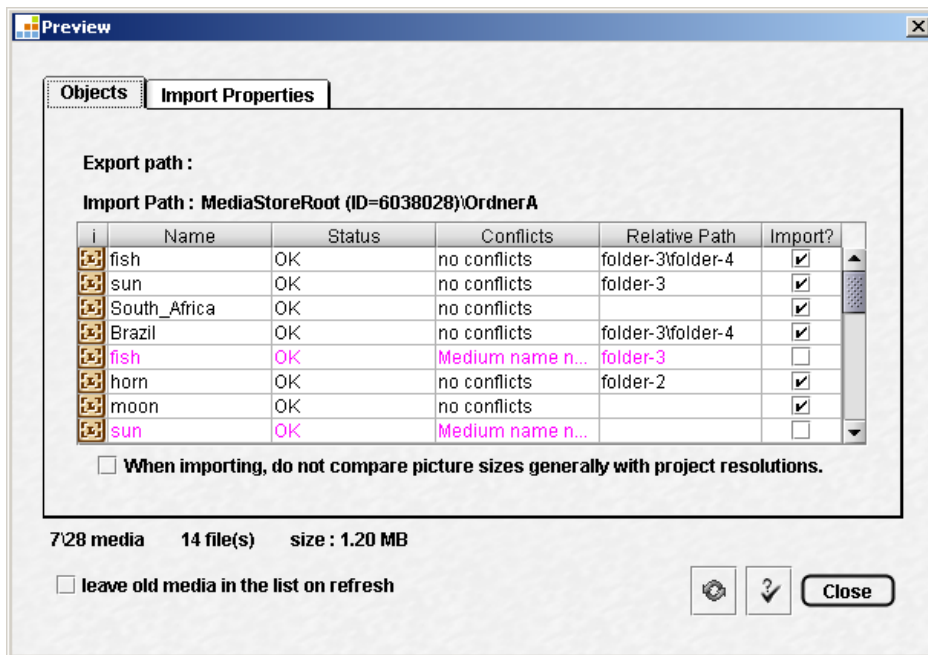## 6.8.9 Media Import Wizard – Preview (Objects tab)

Figure 6-37: Media Import Wizard – Preview – Objects tab

Click the heading fields to display the media sorted by the values in the corresponding column.

Name: This is name as it is to be created in the Media Store.

Status: Indicates whether the import can be completely executed or whether further media are required for a resolution or a language.

Conflicts: States why a medium currently cannot be imported into the Media Store.

Relative Path: Gives the path to the file in the file system relative to the selected export path. If the field is empty the file is directly in the given folder.

Import: Indicates whether a medium has been selected for the import or not. All

selected media are labelled with a tick. Click the box to set or remove a tick.

If a medium is right-clicked a context menu opens:

Remove from list
Rename
Make media names unique
Select All
Invert Selection
import all objects
Import no object
Properties

Figure 6-38: Media Wizard – Preview – Objects tab – Context menu

6.8.9.1   Remove from the list

The selected medium is removed from the list.

6.8.9.2   Rename

The name of the medium can be changed, e.g. to avoid name being issued twice.

6.8.9.3   Make media names unique

If the medium does not have a unique name this function is used to supplement the name so that it is unique.

6.8.9.4   Select all

All the media in the list are highlighted.

6.8.9.5   Invert selection

The selections in the list are reversed. All unselected media are selected and all selected media are unselected.

6.8.9.6   Import all objects

All media without conflicts are selected for the media import, i.e. the tick is set in the

last column.

6.8.9.7 Import no objects

The import selection is undone for all media, i.e. all ticks are removed from the last column.

6.8.9.8 Properties

The Properties window can be used to check which specific file has been selected from the computer's file system for importing for the relevant language or resolution. The settings made can be manually changed if necessary.

6.8.10 Media Import Wizard – Preview (Import Properties tab)

The settings made can be checked again in the Import Properties tab:
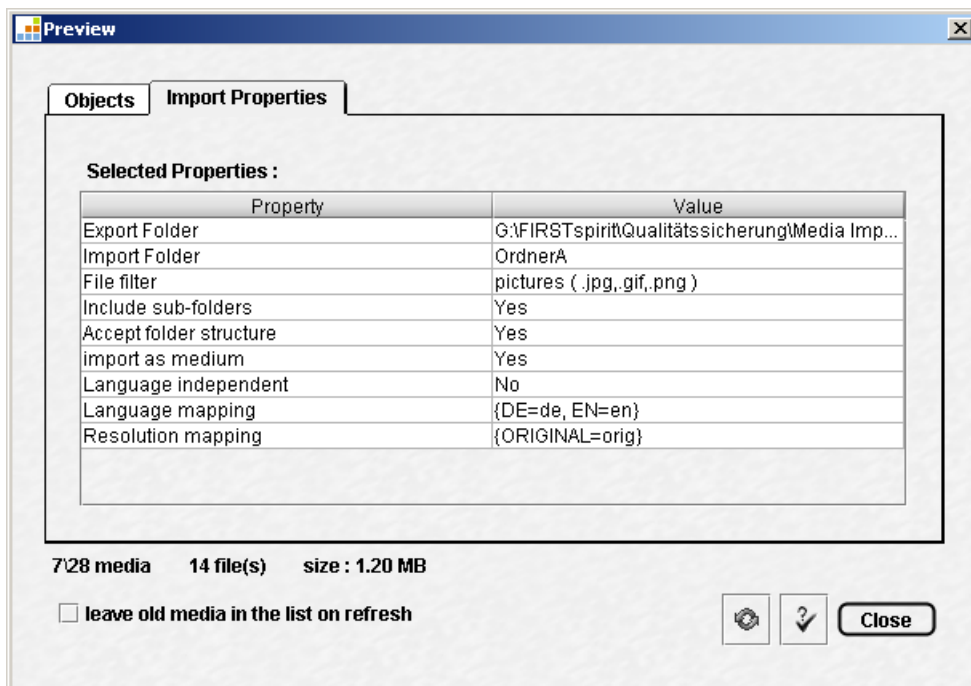

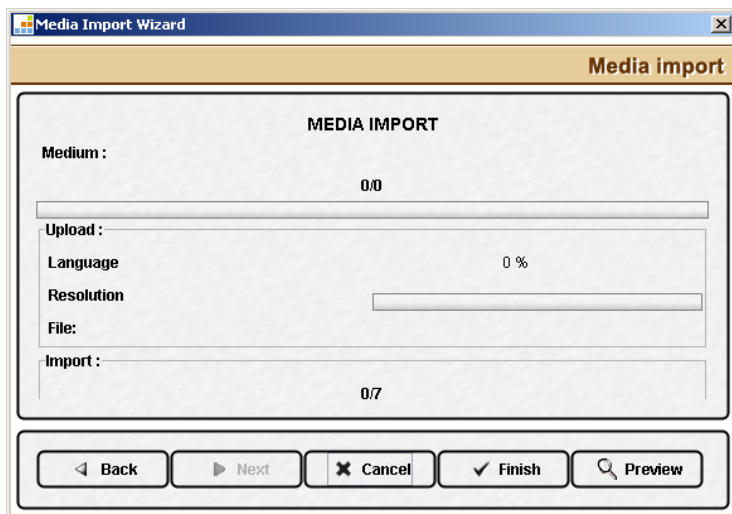**Figure 6-39: Media Wizard – Preview – Import Properties tab**

The preview window can be closed by clicking the Close button.

6.8.11 Media Import Wizard – Step 9 of 9

Figure 6-40: Media Import Wizard – Step 9

If the details in the preview are satisfactory the ✓ Finish button must be pressed as a final step to enable the media to be completely imported into the project's Media Store.

7 Site Store of the JavaClient

Site-Store

The Site Store displays the navigation structure of the website Due to the separation from the layout the appearance and position of the individual navigation levels can be freely defined and changed. Navigation points (including graphical navigation such as JavaScript or Flash) can be added, changed or removed at any navigation depth and at any time. The referential integrity is maintained by Link Management. Each folder in the Site Store corresponds to a menu level in the navigation so that a new menu level is automatically added with each new sub-folder.

The following elements can be created within the Site Store:

- Folders; correspond to a menu level in the website's navigation.
- Start folders; if a menu level does not have a specific page the link is directly forwarded to the pages in the Start Folder.
- Page references; these are the specific pages which can be displayed.
- Start page; if there are several pages in a menu level the start page is displayed first.

- Document group; can group together several page references and menu levels and display them as one page.

7.1 "Drag & Drop" in the Site Store

> **!** *If the function "Confirm move operations" in the menu "Extras" is active, for each move a security dialog box must be confirmed (see Chapter 3.1.5.2 page 78).*

> **!** *The user must have the necessary permissions to "Drag & Drop" (move, copy) from nodes into the Site Store. Otherwise an error message appears "This action is not possible (lack of permissions)!"*

> ![!] *If start pages (![icon]) or start folders (![icon]) are moved you must ensure that there are no longer any start nodes in the relevant higher-level folder. A new start node must be created for the relevant folder, otherwise dead links will be subsequently created in the navigation!*

> ![!] *For a detailed documentation of the enhanced Drag & Drop functionalities from FirstSpirit version 4.2 on see Chapter 11.4 page 318.*

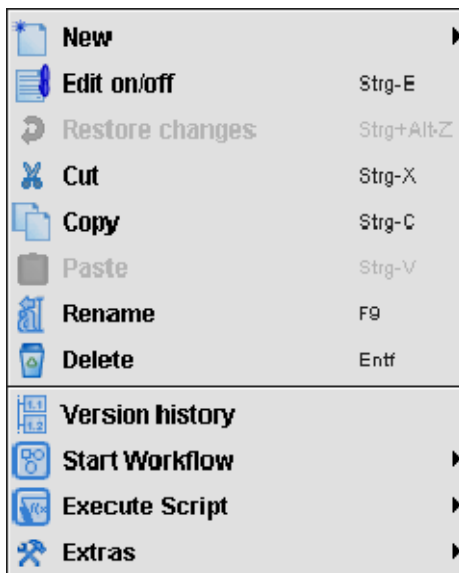## 7.2 General Site Store context menus

Figure 7-1: Context menu – General Site Store functions

The Site Store context menus are described in the following chapters:

For general information on handling context menus see Chapter 4.2 General context menus in the Page Store page 115.

7.2.1 New

The "New" context menu entry can be used to insert new objects into the project. The selection available depends on the object type on which the context menu was opened:
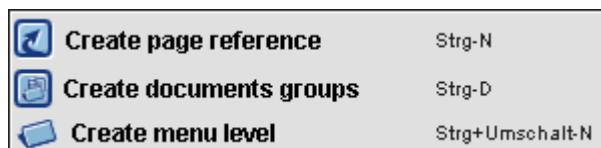
7.2.1.1 New: Root node and Folders

Figure 7-2: Selection at root node or folder level

New – Create Page Reference: Page references must be inserted in the Site Store so that the website's navigation not only consists of various levels but specific pages are also linked in it. After clicking a window opens in which the Page Store is displayed so that you only have to move through this tree until you have found the relevant page. Click this page and the "OK" button to include the page in the navigation structure. The first page inserted in a menu is automatically the start page.

> *It is also possible to create a new page reference by simply dragging the relevant page out of the Page Store and into the required menu folder of the Site Store with your mouse.*

New – Create New Document Group: This function can be used to insert a new document group in the navigation structure. After clicking a window opens in which a unique reference name must be given for the document group. If necessary, it is also possible to give a display name the JavaClient tree structure for each editorial language.

New – Create Folder: This function can be used to insert a new menu level in the navigation structure. Click to open a dialog box in which you can enter the folder name.
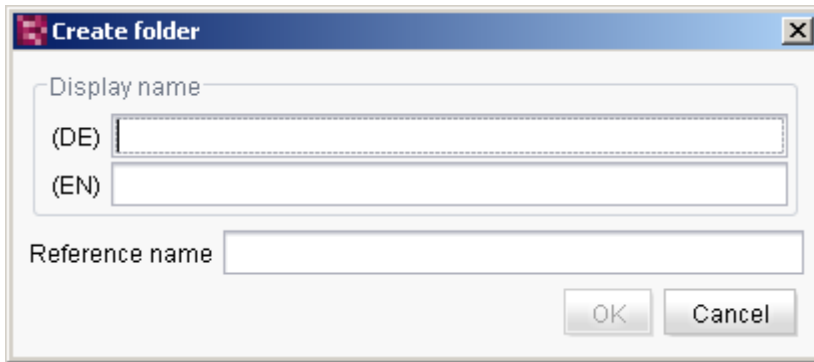
Figure 7-3: New – Create Folder

A **language-dependent display name** can be assigned to the new folder, for each editing language defined by the project administrator. Either the display names or the reference name are then displayed in the tree view, depending on the setting in the "Extras – Preferred Display Language" menu, from FirstSpirit Version 4.2 the "View – Preferred Display Language" menu (see Chapter 3.1.4.2 page 69). The Reference Name field is automatically filled with the value entered for the first display name by the editor, but can be changed (up to the initial creation of the object). The reference name may not contain any spaces, special characters or symbols. This is taken into account accordingly when the field is filled automatically.

> **!** *The Reference Name field is only displayed if this setting was configured accordingly by the project administrator.*

> **!** *In FirstSpirit Version 4.2R4 and higher, the administrator can define rules with which special characters in reference names are automatically transformed into valid characters. The conversion takes place directly during input when creating a FirstSpirit object or changing a reference name (context menu: "Extras" / "Change reference names").*
> *Special characters for which the administrator has not defined a replacement rule cannot be entered in the "Reference name" field.*

> [!] *The name of the menu level in the FirstSpirit JavaClient does not necessarily have to the same as the name of the navigation level on the actual website. How to change the name is described in Chapter 7.5.1.2 page 231.*
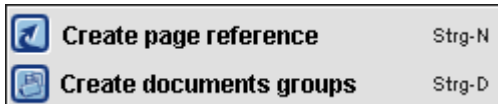
7.2.1.2   New: Page references

Figure 7-4: Selection at page reference level

New – Create Page Reference: After clicking a window opens in which the Page Store is displayed so that you only have to move through this tree until you have found the relevant page. Click this page and the "OK" button to include the page in the navigation structure. The first page inserted in a menu is automatically the start page.

New – Create New Document Group: This function can be used to insert a new document group in the navigation structure. After clicking a window opens in which a unique reference name must be given for the document group. If necessary, it is also possible to give a display name the JavaClient tree structure for each editorial language.

7.2.2   Lock/Unlock (Edit Mode On/Off)

This function is used to activate Edit Mode for the selected node. No other editor can then make changes to this object because it is locked.

Opening this function again deactivates edit mode (unlocks the object), i.e. View Mode is reactivated.

7.2.3   Undo

This function can be used to undo changes made during the current editing process and which have not yet been saved.

7.2.4   Cut

This function is used to cut the current object and store it in the (temporary)

clipboard. It can be inserted again in another position in the tree structure.

7.2.5   Copy

This function is used to generate a copy of the current object and store it in the (temporary) clipboard. Copies of page references can be generated.

7.2.6   Insert

This function is used to insert the content of the (temporary) clipboard in the current position of the tree structure. This function is only active if there is data in the (temporary) clipboard which may be inserted in the current position.

7.2.7   Rename

This function can be used to change the name of the current object in the tree structure of the FirstSpirit JavaClient. After the function is executed a window opens with the object name to date; this can now be changed.

7.2.8   Delete

This function can be used to delete the current object from the tree structure of the FirstSpirit JavaClient. Accidental deletion is prevented by a confirmation prompt.

> *A workflow, e.g. for deleting an object, can be tied to this function from FirstSpirit version 4.1 (see Chapter 3.2.9.7 page 101). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

7.2.9 Version history

A window opens in which all the old versions of the current object are listed. For precise documentation of version management in the Site Store see Chapter 11.12.6 page 411.

7.2.10 Workflow

If workflow is not yet active for the selected object, all workflows defined in the permissions system for these nodes in the tree structure are listed under this menu item. The required workflow can be started under this menu item.

If a workflow is already active for the selected object it can be switched to another workflow action/state under this menu item.

Detailed documentation of workflows is given in Chapter 12 page 422.

7.2.11 Execute Script

All scripts which can be opened in this position in the JavaClient are listed under this menu item. Scripts enable pre-programmed actions or calculations to be executed.

7.3 Special context menus in the Site Store

Refresh this store	F5
Restore deleted elements	

Figure 7-5: Context menu – Special functions at root level

7.3.1 Refresh this Store (at root level)

This menu entry can be used to refresh the view in the Site Store. This is necessary if several people simultaneously work on and make changes to a project.

> ❗ *If you have edited an object and not saved it back to the server you may not use this function! Otherwise the unsaved changes would be overwritten by the server version and would therefore be lost.*

7.3.2 Restore Deleted Objects (at root and folder level)

The Restore Deleted Objects function can be opened both at root and at folder level. If a folder, page or section has been accidentally deleted from the tree structure this function can be used to restore the deleted object. After clicking a window opens with the deleted objects.
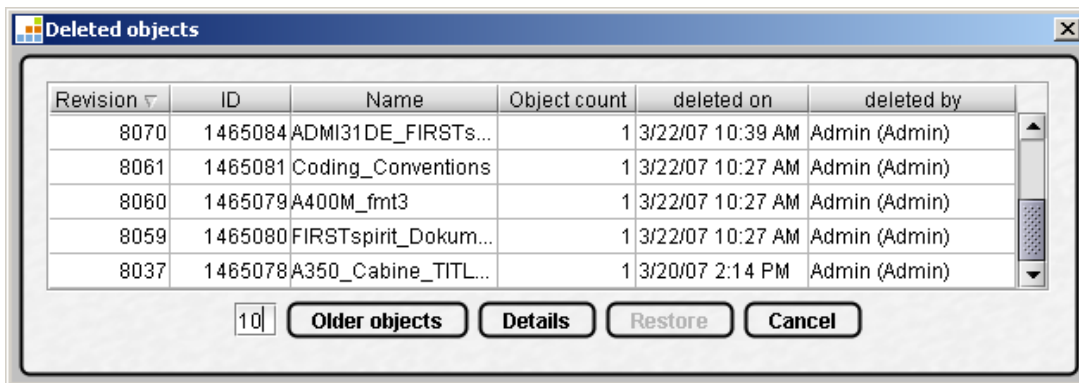

Revision ▽	ID	Name	Object count	deleted on	deleted by
8070	1465084	ADMI31DE_FIRSTs...	1	3/22/07 10:39 AM	Admin (Admin)
8061	1465081	Coding_Conventions	1	3/22/07 10:27 AM	Admin (Admin)
8060	1465079	A400M_fmt3	1	3/22/07 10:27 AM	Admin (Admin)
8059	1465080	FIRSTspirit_Dokum...	1	3/22/07 10:27 AM	Admin (Admin)
8037	1465078	A350_Cabine_TITL...	1	3/20/07 2:14 PM	Admin (Admin)

10 | **Older objects** | **Details** | Restore | **Cancel**

Figure 7-6: Deleted objects

All objects for which a backup exists are displayed at root level, while at folder level only the objects located underneath this folder are displayed. The following information is given for each object:

Revision: Version number of the deleted object.

ID: The unique ID number of the deleted object

Name: The reference name of the deleted object.

Number of objects: The number of objects located in the tree structure below the deleted object. These hierarchically lower level objects are also inserted again by the restore function.

Deleted on: Date and time when the object was deleted.

Deleted by: Name of the user who deleted the object.

Restore To restore, it is only necessary to select the required object and press the button.

The restored objects are, where possible, inserted in the same position in which were located before being deleted. However, if e.g. the folder in which a page to be restored has also been deleted (and is to remain deleted), this page is inserted directly below the tree node at which this function was opened.

> **!** *If on restoring it is found that the referenced page is no longer in the Page Store, this is brought to the user's attention by a new window and they are facade with the choice whether the remaining pages are to be restored or not.*

> **!** *If the process is continued, all page references which cannot be restored are irrevocably lost. If these are to be kept the user is urgently advised to cancel this process and the restore the required pages in the Page Store.*

> **!** *After restoring folders, all the information stored in them (page groups, pictures for the navigation, texts, ...) are also available once more. If abandoned individual page references are restored all information relating to the corresponding page in the opposite folder is lost (e.g. affiliation and position within a page group)*

| Set as default start menu |
| Export |
| Import |
| Restore deleted elements |

Figure 7-7: Context menu – Special functions at folder level

7.3.3 Set as Default Start Menu (folder level only)

This function is intended for transition versions of a website. A start page must usually be declared on each menu level. However, if during the setup of a website the menu levels only are created to set up the structure in the Site Store and the

actual page references are still missing, a sub-level must distinguished for each level by specifying it as the start menu. As soon as page references are subsequently added and one of them is defined as a start page the specification as a start menu is automatically cancelled.

7.3.4 Export (at folder and page reference level)

This function can be used to export the selected folder or the selected page reference with all the necessary information to the hard disk, from where it can later be imported again into another project. A window opens with the file structure of the computer to enable you to specify a suitable storage location for the export.

> **!** *This function can be used to export the selected folder or the selected medium with all the necessary information to the hard disk, from where it can later be imported again into another project. A window opens with the file structure of the computer to enable you to specify a suitable storage location for the export.*

7.3.5 Import (at root and folder level)

This function can be used to import an export file with all the necessary information back into the project. A window opens with the file structure of the computer enabling you to select the required export file.

> **!** *The import function on root level is available only from FirstSpirit version 4.1.*
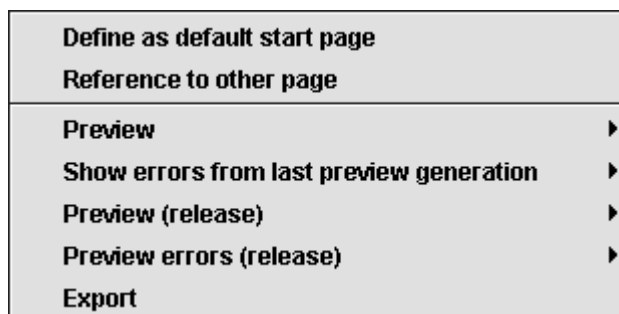
Figure 7-8: Context menu – Special functions at page reference level

7.3.6 Define as default start page (page reference level only)

If there are several page references at a menu level, the FirstSpirit JavaClient must be notified which is the start page for this level, i.e. the page to appear on the website when this navigation item is clicked. If the menu level on which this page is located was previously specified as the start menu, this is automatically undone as soon as the start page is defined.

7.3.7 Reference Other Page (page reference level only)

This function can be used to reference another page from the Page Store. After clicking a window opens in which the Page Store is displayed so that you only have to move through this tree until you have found the relevant page.

7.3.8 Display Preview (page reference and document-level group)

If, while working on an object, you want to check what the finished website will look like you can do this with the help of the Preview function. After entering a browser in the user settings in the Global Store (see Chapter 9.3 page 247), click this button to reference the content of the page or group of documents with the layout and to generate a test version of the website. The page or group of ducuments is then displayed in the selected browser.

This entry shows the current status, i.e. taking into account all the changes just made to the page / documents group (and saved).

The user can continue working with the client during the generation of a preview.

7.3.9 Preview Errors (page reference and document-level group)

If errors occurred during the generation of the preview they can be displayed here.

7.3.10 Display Preview (release) (page reference and document-level group)

This entry shows a preview of the page's current release status, and also the version 4.2R2 document group, i.e. the status last accepted by a person authorised to release.

7.3.11 Preview error (release) (page reference and document-group-level)

If errors occur while the preview is being generated in the release status of the page and in Version 4.2R2 and higher in the document group too, they can be displayed here.

7.4 Administrative context menus in the Site Store

Figure 7-9: Context menu – Administrative Functions

7.4.1 Extras – Release

This function can be used to directly release the current object or a specific release option can be used.

Detailed documentation on the specific release options is given in Chapter 12.4 page 433.

7.4.2 Extras – Delete Metadata

Execute this function to delete all metadata defined for the current level in the tree structure.

7.4.3 Extras – Change Permissions

This function can be used to define the permissions for the current nodes in the tree structure.

Detailed documentation on the definition of permissions is given in Chapter 12.4page 433.

7.4.4 Extras – Reset Write Lock

If a write lock exists for the selected node due to an active workflow the write lock can be cancelled using this function. (The write lock is indicated by italic lettering in the tree)

7.4.5 Extras – View Page (page reference level only)

This function can be used to go to the currently referenced page of the Page Store.

7.4.6 Extras – Show Usages (at folder and page reference level)

This function can be used to determine whether the current folder or the page reference is used in a document group. A window opens in which all uses are listed.

Click one of the uses and the FirstSpirit JavaClient goes directly to the corresponding position in the Site Store.

7.4.7 Extras – Display Properties (from V4.2)

This function can be used to show technical and editorial information on individual project content in a separate dialog and to collate it into a system report. It can also be opened using the keyboard shortcut Alt + P. The information can vary depending on the object type.

For further information about this function see Chapter 4.4.10 page 128.

7.4.8 Extras – Edit Page (page reference level only)

This function can be used to edit the currently referenced page in the Page Store.

7.4.9 Extras – Cancel Editing

Using this function the editing mode can be determined, without accepting changes which have been not saved yet.

7.4.10 Extras – change reference name (at folder, page reference and document-level group , V4.1 and higher)

Each FirstSpirit object has a reference name, which must be unique in each store. Each object can be identified by its reference name.

> **!** *Reference names are only shown in the project if the "Reference names" setting has been activated in the "Extras / Tree Display" menu, in FirstSpirit Version 4.2 and higher: "View – Preferred display language" (see Chapter 3.1.4.2 page 69).*

In general, reference names are issued automatically when objects are created and are based on the display name. These reference names can be subsequently changed using the "Change reference names" menu item. However, the user must have "Change" permission for the respective object. The reference name should only be changed if the object is not yet referenced in the project (e.g. a picture has not yet been used on a page or in a section of the Page Store or a page of the Page Store is not yet used in the Site Store ("page reference")), as otherwise the existing reference becomes invalid. The following dialog is displayed:

Abbildung 7-10: Change reference name

If the reference name is changed despite an existing reference, it may be necessary to restore the reference manually using new selection.

In FirstSpirit Version 4.2R4 and higher, depending on the project administrator's settings, the menu entry may be greyed out regardless of the "Change" permission; in this case the reference name cannot be changed.

7.5 Settings at menu level

After you have added a new menu level in the Site Store using the context menu and have blocked this level you can edit it and therefore extend the website's navigation. The same views and setting options as in the menu levels are available for the root

node of the Site Store. Exception: The root node of the Site Store is however outside the navigation, therefore no navigation settings are made there.

7.5.1 Names tab

The website's navigation can be edited in the Names tab. Setting options are available for graphical and textual navigation.

Figure 7-11: Folders view – Names tab

7.5.1.1 Graphic Navigation

The settings for graphical navigation are set in the top part of the window. Graphical navigation means that the navigation is not based on simple text entered in HTML but some kind of graphics is used instead. These can be buttons with a text or real pictures without any text but symbols or icons clearly understood by the visitor to the site.

Normal: The graphic to be displayed if the menu item is not selected is specified in this field. Click the ![icon] icon to open a window in which the tree structure of the Media Store is displayed.

Figure 7-12: Select Medium

> **!** *A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360).*

After selecting a medium the "media:" appears in the field followed by the name of the selected medium. Click the Delete icon to delete the selected graphic.

> **!** *In the dialog in Figure 7-12 only images can be selected, but no files. From FirstSpirit version 4.1 only images are displayed in this dialog. From FirstSpirit version 4.1 on, the name of the selected image is no more preceeded by "media:".*

With mouse pointer: The graphic to be displayed if the mouse pointer is moved over the menu item is specified in this field.

Selected: The graphic to be displayed if the menu item is selected is specified in this field.

Selected with mouse: The graphic to be displayed if the mouse pointer is moved over the menu item if the menu item has been selected is specified in this field.

These are all the states a graphic can have on a website. If you have defined a picture from the Media Store for all these cases the graphical navigation for this menu item is complete.

> ❗ *As the pictures for the graphic are probably buttons with a text in most cases it must be ensured that language-dependent pictures are used, otherwise the buttons in one language would be displayed in all languages.*

7.5.1.2 Textual navigation

The settings for textual navigation are set in the bottom part of the edit window. Textual navigation means that no graphics are used for the navigation and simple text which can be entered in HTML is used instead.

Menu name: A text which is to represent this menu item within the navigation is entered in this field. This is the name for this navigation item really used on the website, which may differ from the name of the menu level in the Site Star.

Keywords: Several keywords for search functions can be entered in this field.

Comment: This field can be used to enter an explanatory comment on this menu item which, among other things, can be used as a tool tip on the website.

7.5.1.3 General Details

The two bottom options must be used irrespective of whether the navigation is graphical or textural.

Display in the navigation menu?: This option must be activated, otherwise this menu item (including all sub-menus) is removed from the navigation! This option is useful for areas which are to be temporarily removed from the website but re-integrated later. This option saves having to create the structure for this area again.

Display navigation menu in the sitemap?: Activate this option to specify whether this page is to be listed in the sitemap too.

7.5.2 Menu order tab

The Menu Order tab can be used to subsequently change the order in which the menu entries are to appear in the navigation. All menu items below the selected navigation level are listed.
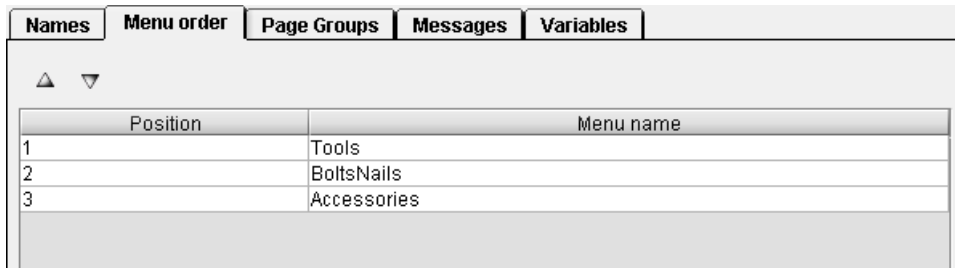
Figure 7-13: Folders view – Menu Order tab

△ Move up; click this icon to move the selected menu level up in the list by one position.

▽ Move down; click this icon to move the selected menu level down in the list by one position.

7.5.3 Page Groups tab

Page groups are a further form of the navigation. They are pages created in the Site Store but which are connected to each other by template programming so that it is possible to page forward and backward in them.

Figure 7-14: Folders view – Page Groups tab

Add Page; click this icon to open a window in which all page references are listed which are located in this menu level and have not yet been inserted in any other page group. You can select which of these pages you wish to include in the group. New pages added are inserted as the last page in a group and therefore appear at the bottom of the list. However, this icon is only active if at least one page group exists.

Delete Page; click this icon to remove the selected page reference from the page group.

△ Move up; click this icon to move the selected page reference up in the list by one

position.

▽ Move down; click this icon to move the selected page reference down in the list by one position.
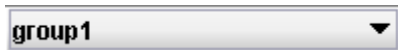
group1 ▼ Select Page Group; here you can select an existing page group for editing.

Create New Group; click this icon to open a window in which the name for the new page group can be entered. Confirm your input with OK to add the new page group.

Delete Group; click this icon to remove the selected page group.

> **!** *If individual pages are to be hidden from a page group (e.g. because they do not have any content in some languages), the tick in front of the "Display in Page Group" entry must be removed. (See Chapter 7.6 on page 235). The result is that the corresponding page is not taken into account in the page group for the respective language. (However, the page is nevertheless generated). If a reference to this page group has been set the top page is referenced.*

7.5.4 Messages tab

The Messages tab contains a message board.

Detailed documentation on use of message boards is given in Chapter 11.2 "The Message Board" page 316.

7.5.5 Variables tab

> **!** *From FirstSpirit version 4.1 the Variables tab is only visible for project administrators to make the editorial work as easy and clear as possible.*

The contents of all input elements of the current page or the current section are stored in variables within FirstSpirit. The "Variables" tab can be used to define variables, with which changes are to be made from subtree to subtree. For example, if a different layout (e.g. a different background colour) is to be used for the individual

sub-areas of the structure, this can be done using the so-called "structure variables".
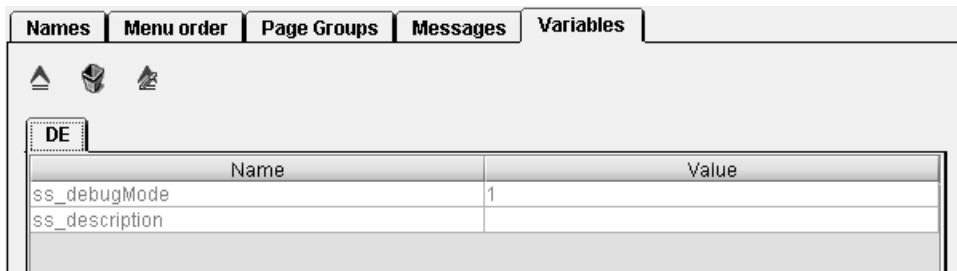
Figure 7-15: Folders view – Variables tab

⬜ Create New Variable; click this icon to open a window in which the name for the variable can be entered. Confirm your input with OK to add the new variable; a window for editing the new variable automatically opens.

🪓 Delete Variable; click this icon to remove the selected variable.

✎ Edit Variable; click this icon to open a window for editing the selected variable.
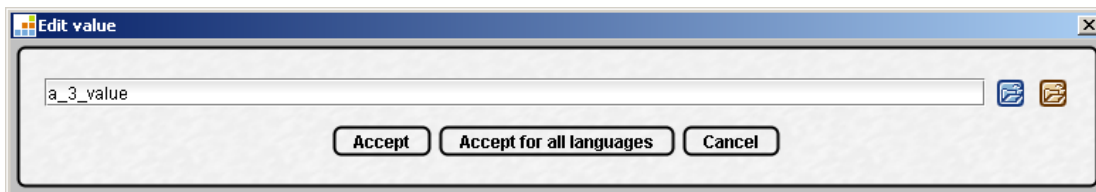
Figure 7-16: Edit variable

In this window you can enter the value of the variable manually or you can select a page reference 📂 or as a medium 📂 value.

Accept Click this button to accept the selected variable value for the current language.

Accept for all languages Click this button to simultaneously accept the selected variable value for all project languages.

> ❗ *The variables must be referenced within a template in order to cause an effect. More precise information on programming templates is available in the online documentation.*

> ❗ *If the value of a variable is changed without releasing the Site Store and then a preview of a page is requested from the Page Store and this variable is valid for its page reference the current status (i.e. not released) of the variable is displayed.*

7.6 Settings at page reference level

Edit Mode is automatically activated after a new page reference is added to the Site Store. The page can now be edited and the website's navigation therefore extended.

7.6.1 Languages tab

A separate tab is created for each project language.

Figure 7-17: Page Reference view – Languages tab

- General Details   (see Chapter 7.6.1.1 page 236)
- Menu Overview (sitemap)   (see Chapter 7.6.1.2 page 236)
- Link   (see Chapter 7.6.1.3 page 237)
- Validity period   (see Chapter 7.6.1.4 page 238)

7.6.1.1 General Details

File Name: The file name under which the referenced page is stored on the server is entered in this field (see Figure 7-17).

> *The name can be changed; however, you must ensure you use a name valid for the server's file system. You should also ensure that the file extension satisfies the relevant conventions. The characters allowed for the file name are: abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ-_*

Show in Page Group: Activate this option to display the current language of the page reference in a page group. This option must be deactivated if the page is not to be displayed in a page group for the current language.

Keywords: Several keywords for search functions can be entered in this field.

7.6.1.2 Menu Overview (sitemap)

Show in Sitemap: If this option is activated the current menu level is displayed in a sitemap (see Figure 7-17).

Text for Sitemap: This field can be used to enter a text which is to be displayed in a sitemap for this menu level. This text is also used in the output of tables of contents of page groups. If no text is entered here the page name from the Page Store is used in both cases (see Figure 7-17).

Picture for Sitemap: This field can be used to select a picture which is to be displayed in a sitemap for this menu level. Click the icon to open a window in which the tree structure of the Media Store is displayed.

Figure 7-18: Select Medium

After selecting a medium the "media:" appears in the field followed by the name of the selected medium. Click the Delete icon [X] to delete the selected graphic (see Figure 7-17).

> [!] *From FirstSpirit version 4.1 the name of the selected medium is no more preceeded by "media:".*

7.6.1.3 Link

One of three alternatives for the page reference can be selected in this area (see Figure 7-17).

Internal (FirstSpirit): This option is the default setting; all pages are generated as usual.

Use external URL: If a page from the project is not displayed for some reason or another, activate this option and enter a reference to the page beginning with "http://" in the field; this is the page to be displayed if someone clicks this navigation item on the website.

Use URL from related project: FirstSpirit supports remote access to other FirstSpirit projects. Activate this option to reference to a page from a related project.

[X] Click on this icon to remove entered projects form this field.

[icon] To this end the required project is selected first by clicking the icon. The dialog

with the Site Store of the related project opens. Here, the required reference can be selected. If more than one remote project is available you can switch to them by using the tabs (see Figure 7-19).

> **!** *Only remote projects for which the project administrator has activated the "related projects" function are displayed on the tabs.*

> **!** *The "related projects" function is a licence-dependent additional module (for further information, see Chapter 11.14.1 page 417).*
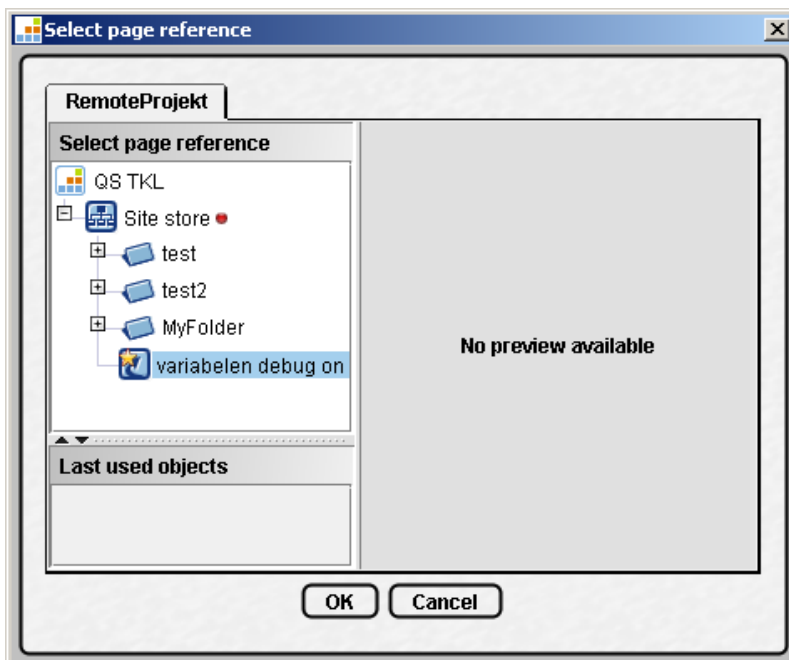
Figure 7-19: Select reference from the remote project

The selected page reference from the remote project is imported as a link in the page reference of the current project. When the respective menu entry is clicked the user is directly switched to the report project page.

7.6.1.4 Validity period

If another page is to be referenced from the Page Store from a specific date, this can be specified at this point. Right-click to open a context menu (see Figure 7-17):

New
Delete entry

Figure 7-20: Context menu in Timed Link area

New: Select this context menu entry to open a window in which the tree structure of the Page Store is displayed. After a page has been selected for the future link another window opens for the date selection (see Chapter 10.7 page 265). This can be used to specify the date from which the currently reference page of the Page Store is to be replaced with the new page.

Delete Entry: Select this context menu entry to remove the selected link from the list.

7.6.2   Content tab

7.6.2.1   General Details

This tab can be used to set the settings for the output if the selected page reference is based on a page from the Page Store with integrated data source.

Figure 7-21: Page Reference view – Data tab

Number of entries per page: Use this field to specify how many data records from the database are to be displayed on each page. As soon as the required number of

entries is reached a new page is created on which the following rows of the database are generated. The default value for this field is 0, whereby all the rows of the data source are displayed on one page.

Maximum number of pages This field can be used to specify the maximum number of pages to be generated. If the configuration of the data results in the generation of more than one page all the generated pages are automatically integrated in a new page group so that they can be referenced using the page group functions. (See manual for developers on creating templates)

> *If the value 0 is entered as the maximum number of pages there is no upper limit for the generation of pages.*

Page for empty query result: If, due to the settings made, no data records can be output (i.e. the query is empty), use the [icon] icon in this field to select a page from the Page Store which is to be referenced instead. This way it is possible to prevent the generation of pages without meaningful content.

Variable for sitemap text: Click the [icon] icon to select a column from the data source. The content of this column is then displayed in a sitemap.

7.6.2.2   Limit the number of data records

Select a query: The combo box can be used to select a query predefined by the project developer to limit the output of the data records.

All parameters specified for the selected query are listed in the table below. The given values of a parameter can be changed by double-clicking the relevant row.

7.6.2.3   Sort

Apart from limiting the number of data records it is of course also possible to sort the output. Any table column can be used as the sort key. The output can be in ascending or in descending order.

[icon] Add Sort Rule; click this icon to use a selection box to specify another table column as the sort rule.

[icon] Delete Sort Rule; click this icon to delete the activated sort rule.

⬘ Move up; click this icon to move the selected sort rule up in the list by one position.
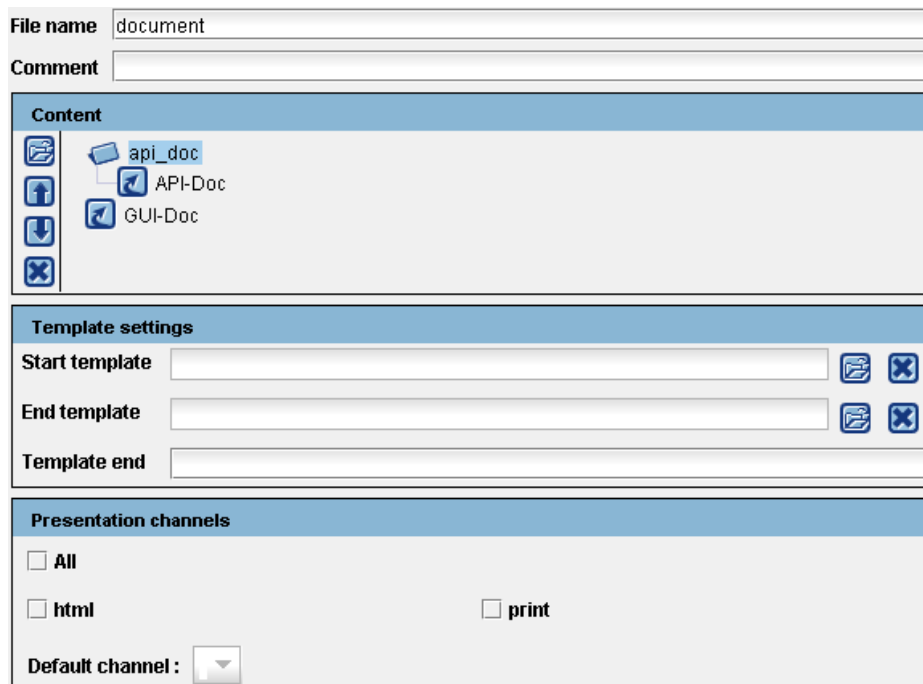
▽ Move down; click this icon to move the selected sort rule down in the list by one position.

If several sort rules are specified the top sort rule in the list is used for the sort order first. All entries for which this first sort rule is identical are then sorted by the second sort rule, whereby the overall first sort is retained. The procedure is the same for each additional sort rule.

7.7 Document group settings

Document groups are special Site Store elements which can be selected as a reference target but do not appear in the navigation.

Both page references and Site Store menu levels can be grouped together in a document group and displayed as a page. If a new page group is added one menu level later, this change is automatically adopted in the document group.

Figure 7-22: Document Group view

7.7.1 General Details

File Name: The file name under which the document group is stored on the server is

entered in this field.

Comment: A comment can be entered in this field. This comment can represent the title heading in the generated document (e.g. PDF).

7.7.2   Contents

The constituents of the document group can be defined in this area.

 Click this icon to insert new elements in the document group.

 Use this icon or the "Del" key to remove selected elements again.

 Move up; click this icon to move the selected content up in the list by one position.

 Move down; click this icon to move the selected content down in the list by one position.

> *If menu levels are included in the document group these menu folders can be opened. However, this is a pure view function; the objects in this menu folder cannot be changed.*

7.7.3   Template settings

This area is used to define which template is to be displayed before or after the document group. This way structures which only appear once (e.g. lists of content) can be generated.

Start template: The icon can be used to select a template which forms the frame for a valid document in the selected presentation channel. The start template forms the "header" of the document.

End template: The icon can be used to select a template which forms the frame for a valid document in the selected presentation channel. The end template forms the "footer" of the document.

Template suffix: This field can be used to define the ending with which the templates to be used instead of the original templates of the integrated pages are

labelled. These templates are used solely for display within the document group.

7.7.4   Presentation channels

All: If this option is activated all available presentation channels are taken into consideration in the generation.

Selected (html, PDF): One or several presentation channels to be taken into account in the generation can be activated here. If one presentation channel only is activated the document group is also generated for this one presentation channel only (e.g. PDF).

Standard channel: This combo box can be used to select one of the activated presentation channels. If only one presentation channel is active this field is highlighted in grey. If a reference is made to this document group at a later date this presentation channel is automatically displayed, provided the reference does not explicitly specify another presentation channel.

8 Template Store of the JavaClient

Template-Store

The internet presence basics are created and managed in the Template Store. The templates of the Template Store take into account the complete design of the website (including Corporate Design and Corporate Identity). The templates are required to joint the contents entered in the Page Store and the media integrated in the Media Store with the structure deposited in the Site Store to form a complete presentation when the website is generated.

Templates are a basic framework for the presentation of contents. I.e. they give the contents of a website the appropriate appearance. Different types of templates are available to template developers in FirstSpirit.

Page Templates create the basic framework of a page and are used to define what the whole internet presence will look like, e.g. where logos and navigation are to be position, whether a page is to be made up of frames and similar general settings. In addition, page templates are used to specify where an editor can insert contents.

Section Templates are used to insert contents in this basic framework. A section template is divided into individually specified input fields which the editor uses in the Page Store to enter the contents of the section. If selected section templates only are to be made available to an editor within a specific page template, this can be defined using **Restrictions**.

Format Templates are used to more conveniently design the text entries in the default input elements, as in e.g. the DOM Editor. Each define d format template is then available in the DOM Editor for formatting the text entries. Table format and style templates are special templates which are used for formatting so-called inline tables (see Chapter 11.6.10 page 351) zur Verfügung stehen.

The appearance of references is specified in detail in the **Link Templates**. Both the input screen forms via which the editor can enter the relevant data and the presentation of the references to the website.

Scripts can be used to automate different types of operating sequences / user input sequences in FirstSpirit. A script is used to describe the sequence to be executed and if necessary can make changes to the FirstSpirit data structure. Scripts can be executed in any point of the client for which they are approved.

Database Schemata define the structure of highly structured databases and the layout for the corresponding data entry screen form. The tables and relations of a data model are depicted in a schema. The input elements for the table columns are then defined in the corresponding table templates and constraints for the databases are formulated in the queries.

Workflows can be used to describe processes or work sequences and can be depicted in a model – the so-called workflow model. The task of the workflow model is to describe the workflow abstractly but also completely so that it can be used as the basis for supporting the user when the carry out a work process.

> !   *The creation and changing of templates requires basic knowledge of HTML and XML and is rarely the task of a "normal" user or editor. Therefore, this document does not deal with template development with FirstSpirit in any greater depth. An instruction and reference document with all the functions and input components will be issued separately.*

A detailed description of the Template Store is given in the "FirstSpirit Manual for Developers (Basics)" documentation.

A reference document with all functions and input components is included in our online documentation.

9 Global Store

Global-Store

The Global Store (global settings) is divided into three areas.

The **Global Content Area** is frequently used for maintaining smaller page components, e.g. Copyright notices.

In **Project Settings** a project developer can specify rules for content replacement.

In the **User Settings** each user can integrate editors for editing specific media and different browsers for the preview function.

9.1 Global Content Area

The context menus and method of working in Global Content Area correspond to the context menus and method of working in Page Store. The standard input elements for pages and sections are also available here for entering contents.

The following elements can be created within the Global Content Area:

- Folders for structuring the pages.
- Global Pages
- Sections

Detailed documentation of **General Context Menus** is given in Chapter 4.2 page 115.

Detailed documentation of **Special Context Menus** is given in Chapter 4.3 page 121.

Detailed documentation of **Administrative Context Menus** is given in Chapter 4.4 page 125.

Detailed documentation of **Standard Input Elements** is given in Chapter 10 page 249.

9.2   Project Settings

The global settings for the website which are to appear on each page can be set in the project settings. Settings can only be made here if the administrator selected a page template in the server and project administration application in which the global project settings are to be defined. The settings made in the relevant Form tab (GUI.XML) are executed here. The input elements can be used to define the global design of the websites. In analogy to the maintenance of the editorial content, the Project settings can be applied language-dependent, i.e. individually for each project language

9.3   User settings

9.3.1   Browser tab

Before using the Preview function in the Page or Site Store you should set the browser in which you want to view the preview. Several browsers for viewing preview pages can be entered in the Browser tab.
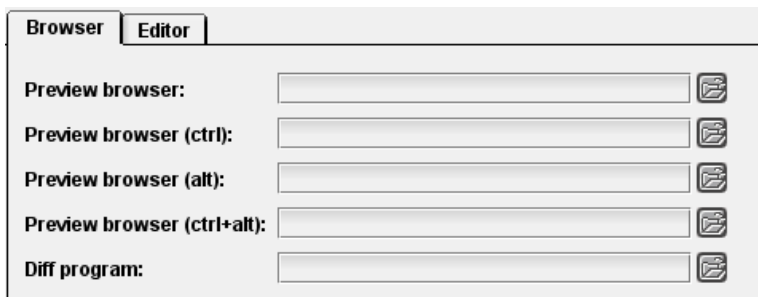
Figure 9-1: User Settings view – Browser tab

Browser: Enter the default browser to be used for each preview in this field. Click the ⬚ icon to select the required browser. A window opens in which you can move through the file structure on your workstation to search for the required exe file.

Further browsers can be specified in the **Browser (Ctrl)**, **Browser (Alt)** and **Browser (Ctrl + Alt)** rows. Open the preview with the corresponding Ctrl and/or Alt key to display the preview using the browser entered there.

Diff Program: A program for comparing text files can be entered in this field. However, this function is of interest for template developers only; an editor will not

usually need it.

9.3.2   Editor tab

Before you can e.g. edit the media in the Media Store you must set which programs you want to use to do this. Suitable editors for the different file extensions can be entered in the Editor tab.
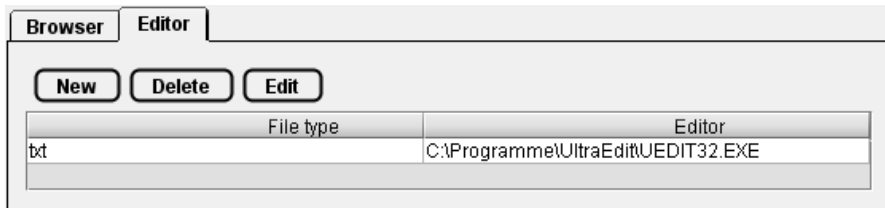
Figure 9-2: User Settings view – Editor tab

New Click this button to open a window in which the new file type and the corresponding editor can be specified.

Figure 9-3: Edit File Type

Click the ⬚ icon to select a suitable editor from the workstation's file structure.

Delete Click this button to remove the selected editor from the list.

Edit Click this button to open a window where a new editor can be specified for the selected file type.

10 The Standard Input Components

10.1 Section selection (CMS_INPUT_SECTIONLIST)

CMS_INPUT_Sectionlist	
☐ CMS_INPUT_DOM	DOM-Editor
☐ CMS_INPUT_SECTIONLIST	Absatzauswahl
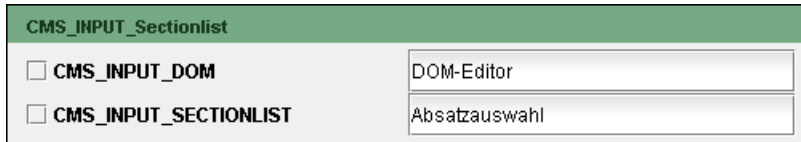
Figure 10-1: Input component – Section selection

This input component can be used to select individual sections of a page for display. According to the setting under **Extras / Preferred display language / Display reference names in tree**, from FirstSpirit Version 4.2 under **View / Preferred display language / Display reference names in tree** the existing sections are displayed with their reference names if the option Display reference names in tree is activated. If not, the display name is used. Simply click the relevant section; the selected sections are denoted by a tick. A name can be assigned for each section and is used for the output.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

10.2 Section list (CMS_INPUT_CONTENTAREALIST)

CMS_INPUT_CONTENTAREALIST, 'label', '*'	
section_1	
	section_2
section_3	
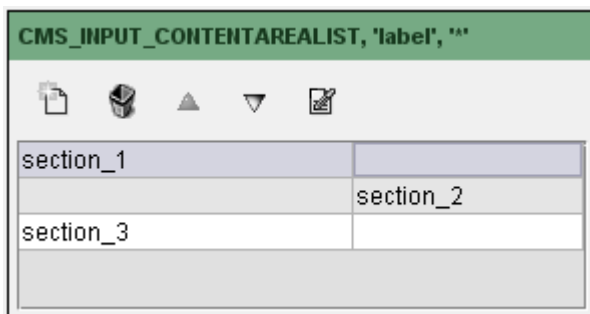
Figure 10-2: Input component – Section list

This input component is used to create, edit and delete sections. Each section has a symbolic name which is assigned when it is created, a section template which is also selected by the editor when creating the section and contents which can be entered

in the relevant input components depending on the template used. The symbolic names of the sections can be displayed instead of the values depending on the configuration of the input component.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

Delete Section; click this icon to delete the selected section from the section list.

Move Section Up; click this icon to move the selected section up in the list by one position.

Move Section Down; click this icon to move the selected section down in the list by one position.

Add Section; click this icon to add a new section to the list. A further dialog window opens which is used to define the new section.

First, a section template to be used for the new section must be selected:
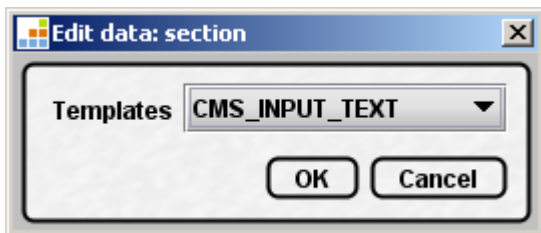
Figure 10-3: Select template

A language-dependent name for the new section can then be assigned in the following dialog.

Figure 10-4: Enter language-dependent name

✎ Edit Section; click this icon to open a window in which the contents of the selected section can be edited.

Figure 10-5: Edit section

10.3 Picture selection (CMS_INPUT_PICTURE)
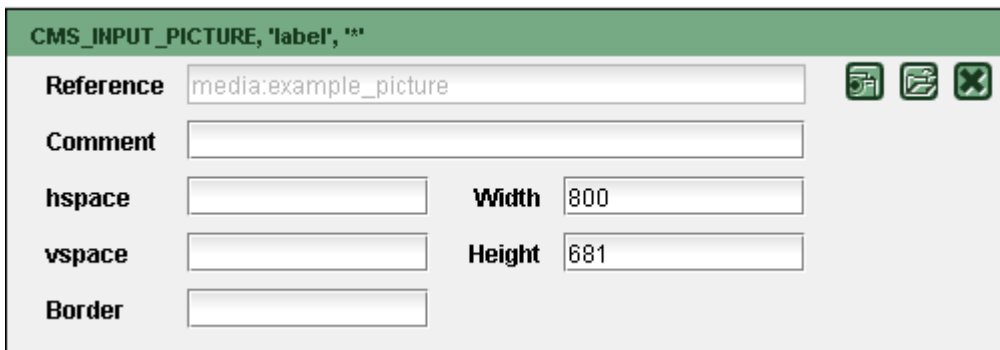
Figure 10-6: Input component – Picture selection

The picture selection enables the entry of pictures in a page or a section in the Page Store. Depending on the configuration and the access permissions, this input component can be used to reference picture files from the local Media Store and/or the Media Store of a remote project (see Chapter 10.3.1 page 253).

If the project developer has defined a preset default value for the input component the ⊡ "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

Reference: A reference to the required picture from the project's Media Store (or of a remote project, from FirstSpirit version 4.2 of several remote projects too) is entered here. The buttons at the end of the input field are used to select a picture or to delete an already existing reference. The buttons can be used as follows:

⊡ Click the icon to open a window in which the Media Store tree structure is

displayed.
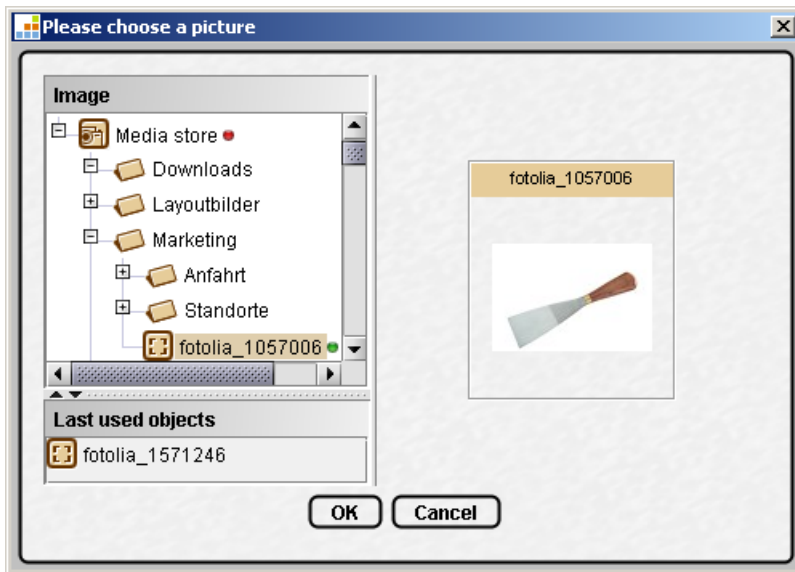
Figure 10-7: Select picture

> ! The media selection of this component is solely limited to all media created as a picture (and not as a file) in the Media Store. (For file selection details, see CMS_INPUT_FILE page 273).

> ! A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the Look & Feel "Classic" the picture selection works as follows.

The required picture can be selected from the tree structure. When a medium is clicked a picture preview (thumbnail) appears in the right-hand side of the window to make the picture selection easier. Click a media folder and a preview of all pictures in this folder appears. Below the tree structure there is also a list of the most recently used objects of the respective user; this is intended to make it easier to find frequently used objects.

This icon is displayed depending on the configuration of the input component. If the component allows uploading of media into the project's Media Store, click the this icon to open a dialog window for selecting a target folder of the Media Store. (If no folders exist in the Media Store, the root node can be selected as upload target.)

Depending on the configuration of the input component, the upload can also take place in a remote project (see Chapter 10.3.2 page 255).

- Create language-dependent medium   (see Chapter 10.3.3 page 258).
- Create language-independent medium (see Chapter 10.3.4 page 261).

Click the icon to switch to the selected medium in the Media Store.

Click the icon to remove the selected medium from the input component.

Comment: If required a comment on the picture can be entered here.

hSpace: Distances to the surrounding text on the left or right-hand side are defined by manually changing the "hspace" value.

VSpace: Distances to the text above and below the picture are defined by manually changing the "vSpace" value.

Border: A border around the picture can be added with a number not equal to 0; the thicker the border the larger the number.

Width, Height: The values for width and height are automatically accepted.

10.3.1   Selection of media from a remote project ("Remote Media")

> ! *The "Remote Media" function is a licence-dependent additional module.*

Picture files are selected from the remote project using a specially configured picture input component. Depending on the configuration and the access permissions, this input component can be used to reference picture files from the local Media Store and/or the Media Store of the remote project. The display of the input component for picture selection is identical to that of the local input component (see Chapter 10.3 page 251).
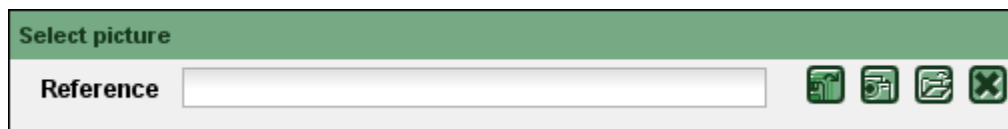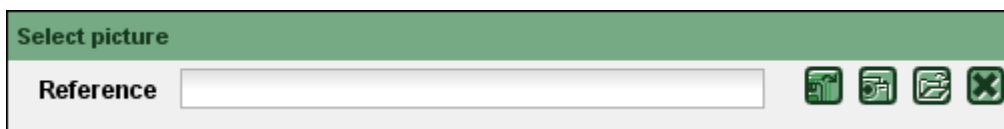

Figure 10-8: Input component for the selection of media

Click the "Select Picture" icon to open the "Please select a picture" dialog

window. Depending on the configuration of the input component:

- the project's local Media Store only is displayed.

- The Media Store of one or several remote projects only is displayed.

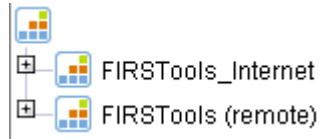- Both Media Stores (remote and local) are displayed (see Figure 10-9).
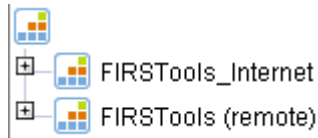
Figure 10-9: Local project and remote project selection

> ❗ *From FirstSpirit version 4.2 the Media Stores of **more than one** remote project can be displayed.*

The required Media Store can be expanded in the dialog window. A picture file is then selected from one of the Media Stores (see Figure 10-10). For further general information on the "Picture Selection" input component, see Chapter 10.3 page 251.

> ❗ *If the picture was selected via the remote project's Media Store it only exists in the target project via this reference, i.e. it cannot be found in the target store's Media Store.*
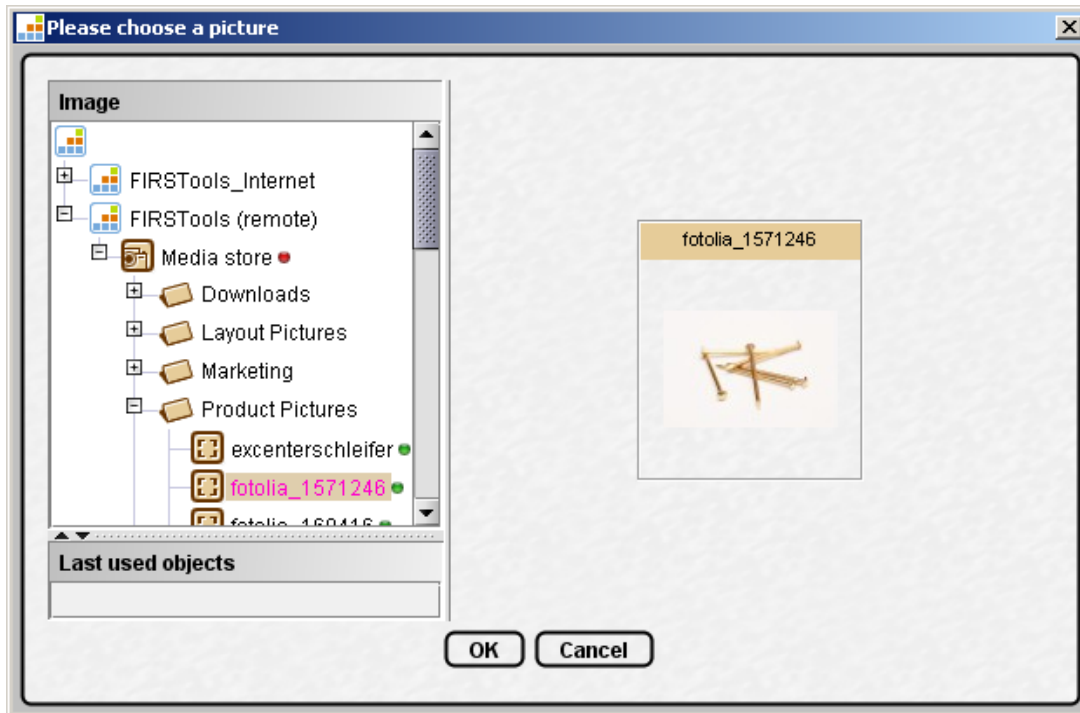
Figure 10-10: Picture selection from two Media Stores (local & remote)

10.3.2 Uploading media into a remote project ("Remote Media")

> ! The "Remote Media" function is a licence-dependent additional module.

Picture files are created in the remote project using a specially configured picture input component. Depending on the configuration and access permissions, this input component can be used to upload picture files from the local file system into the Media Store of the remote project. The display of the input component for picture selection is identical to that of the local input component (see Chapter 10.3 page 251).

Figure 10-11: Input component for the selection of media

Click the "Upload Picture" icon to open the "Please select the upload folder"

dialog window. Depending on the configuration of the input component:

- the project's local Media Store only is displayed.
- The Media Store of one or several remote projects only is displayed.
- Both Media Stores (remote and local) are displayed (see Figure 10-21).

Figure 10-12: Local project and remote project selection

> ! *From FirstSpirit version 4.2 the Media Stores of **more than one** remote project can be displayed.*

The required Media Store can be expanded in the dialog window. A folder is then selected as the target folder for the uploading of the medium from one of the Media Stores (see). Depending on the configuration of the input component, it is possible that only one target folder in the remote project is released for uploading media.

For further general information on the "Picture Selection" input component, see Chapter 10.3 page 251.

> ! *New media objects can be created in existing folders only. Directories may not be created in the Media Store of the remote project.*
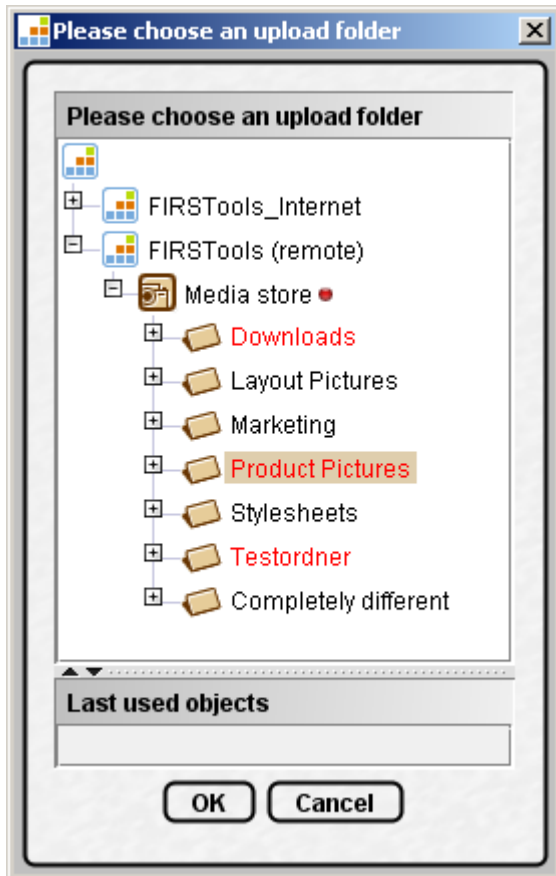
Figure 10-13: Picture upload into two Media Stores (local & remote)

Cancel Click the program to exit the function. No upload folder is selected.

OK Click the button to define the selected folder as the target folder. In the next step the "Create Language-Dependent?" dialog window opens.

▪ Create language-dependent medium (see Chapter 10.3.3 page 258).
▪ Create language-independent medium (see Chapter 10.3.4 page 261).

> **!** *The upload of language-dependent media into the Media Store of the remote project is supported if all the languages of the target project used are also available in the remote project.*

10.3.3 Create language-dependent media
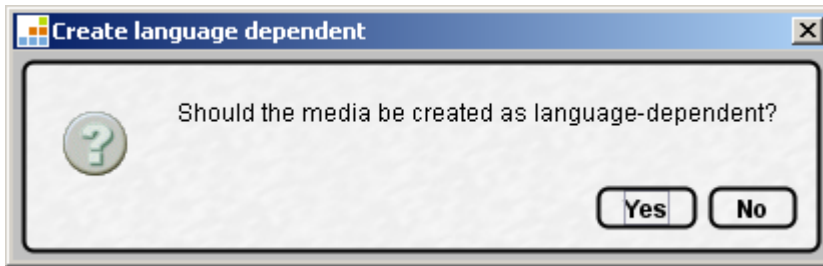
Figure 10-14: "Create Language-Dependent" dialog

Yes Click the button to create the picture language-dependent, i.e. a different picture is displayed for each language in the project and the same reference.

> **!** *The creation of language-dependent media can be prevented via the input component in the Template Store. In this case the "Create Language Dependent" dialog is not opened and the medium is automatically created language-independent.*

A dialog window is opened for selection of the picture file(s):
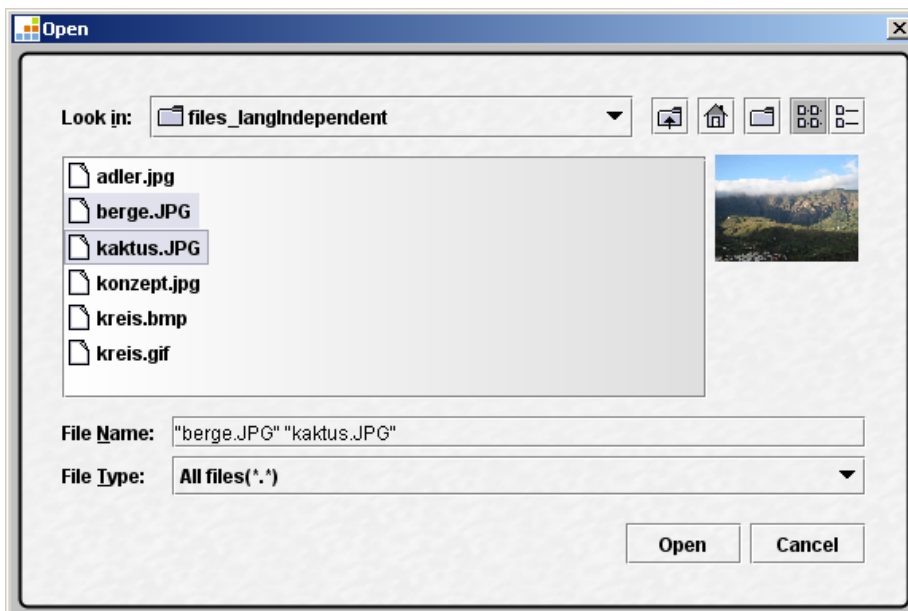
Figure 10-15: File selection

> ❗ *According to the settings in the project configuration (see FirstSpirit Manual for Administrators), the selection of media can be restricted to specific file sizes and file types from FirstSpirit version 4.1 (see Chapter 6.2.1.1 page 162). Please bear in mind, that always the settings of the respective project are valid, i.e. when uploading media to a remote project the media restrictions of the remote project are valid, when uploading media into the target project the restrictions of the target project are valid.*

Precisely the same number of pictures as available project languages should be selected for language-dependent creation of a picture. Multiple selection is possible by pressing the Ctrl key.

In the next step the selected media can be assigned to the required project language.
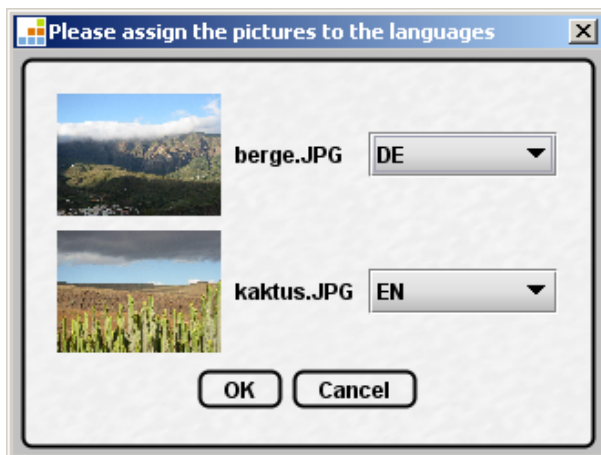
Figure 10-16: Assign selected pictures to the project languages

Cancel Click the program to exit the function. No new picture is created in the upload folder.

OK Click the button to create the selected pictures language-dependent in the upload folder of the remote project or the target project (depending on the selection). A unique reference name under which the medium can be referenced in the project must be given first.
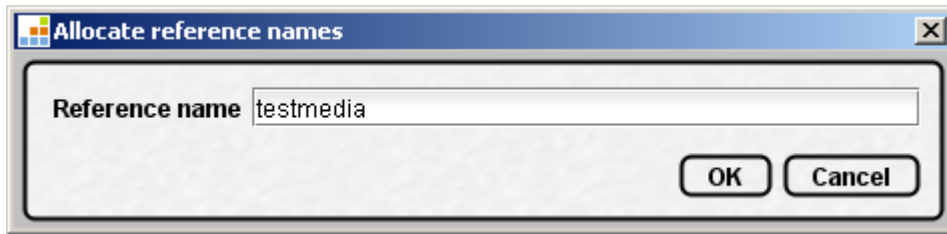
Figure 10-17: Assign reference name for a medium

> ❗ *In FirstSpirit Version 4.2R4 and higher, the administrator can define rules with which special characters in reference names are automatically transformed into valid characters. The conversion takes place directly during input when creating a FirstSpirit object or changing a reference name (context menu: "Extras" / "Change reference names").*
> *Special characters for which the administrator has not defined a replacement rule cannot be entered in the "Reference name" field.*

An optional description of the medium can then be assigned for each project language.

Figure 10-18: Optional description of a medium

The selected pictures are imported into the input components and are displayed in the respective language with a thumbnail. The "Reference" field now contains a reference to the new media object created.

If the picture was created in the remote project and this project "Uses Releases" the new picture created can be released automatically (depending on the configuration of the input component).

10.3.4 Create language-independent media
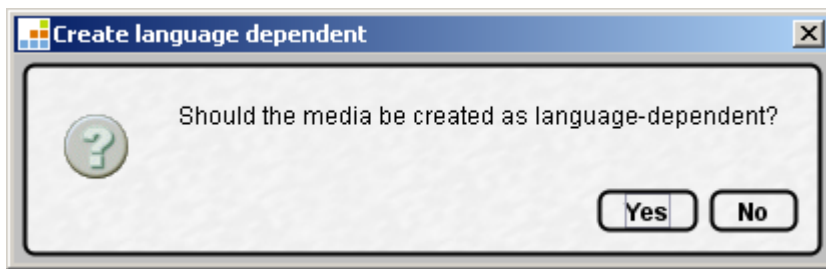
Figure 10-19: "Create Language-Dependent" dialog

> **!** *The creation of language-dependent media can be prevented via the input component in the Template Store. In this case the "Create Language Dependent" dialog is not opened and the medium is automatically created language-independent.*

No If the button is clicked the picture is created *not* language-independent, i.e. the same picture is entered in the input component displayed for each language in the project. Unlike language-dependent creation of a picture only one file can be selected in the following "Open" dialog.

The further procedure is analogous to language-dependent creation of a medium (cf. Chapter 10.3.3 page 258).

10.4 Checkbox (CMS_INPUT_CHECKBOX)

Figure 10-20: Input component – Checkbox

The checkbox provides the user with an option for selecting inputs defined by the project developer in the Template Store. Unlike combo boxes and radio buttons with checkboxes it is possible to simultaneously make more (or less) than one selection.

If the project developer has defined a preset default value for the input component the [icon] "Accept Default Value" icon is located in the top right-hand corner. (For

further information, see Chapter 11.10 page 386).

10.5 Combo boxes (CMS_INPUT_COMBOBOX)
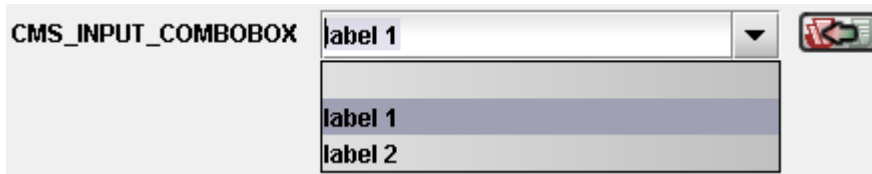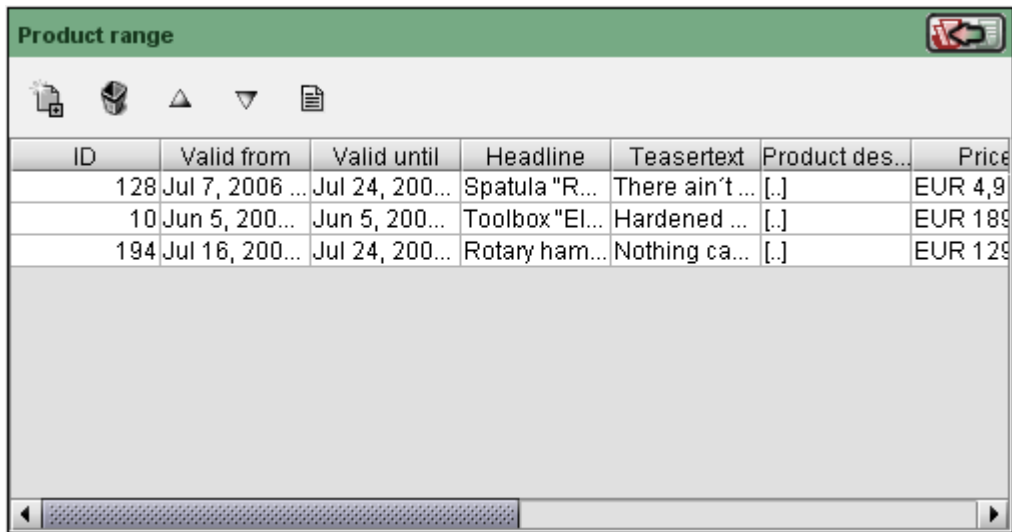
Figure 10-21: Input component – combo box

The combo box provides the user with an option for selecting inputs defined by the project developer in the Template Store. The input is selected by clicking the arrow symbol behind the input row. The selection list then opens, from which the required input can be selected. Precisely one selection is always possible with the combo box.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located on the right next to the component. (For further information, see Chapter 11.10 page 386).

10.6   Data Record Selection List (CMS_INPUT_CONTENTLIST)

Figure 10-22: Input component – data record selection

This input component can be used to create a list of individual data records from a data source.

If the project developer has defined a preset default value for the input component the ![icon] "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

![icon] Add Data Record; click this icon to open a window with the table view of the data source as found in the Content Store.
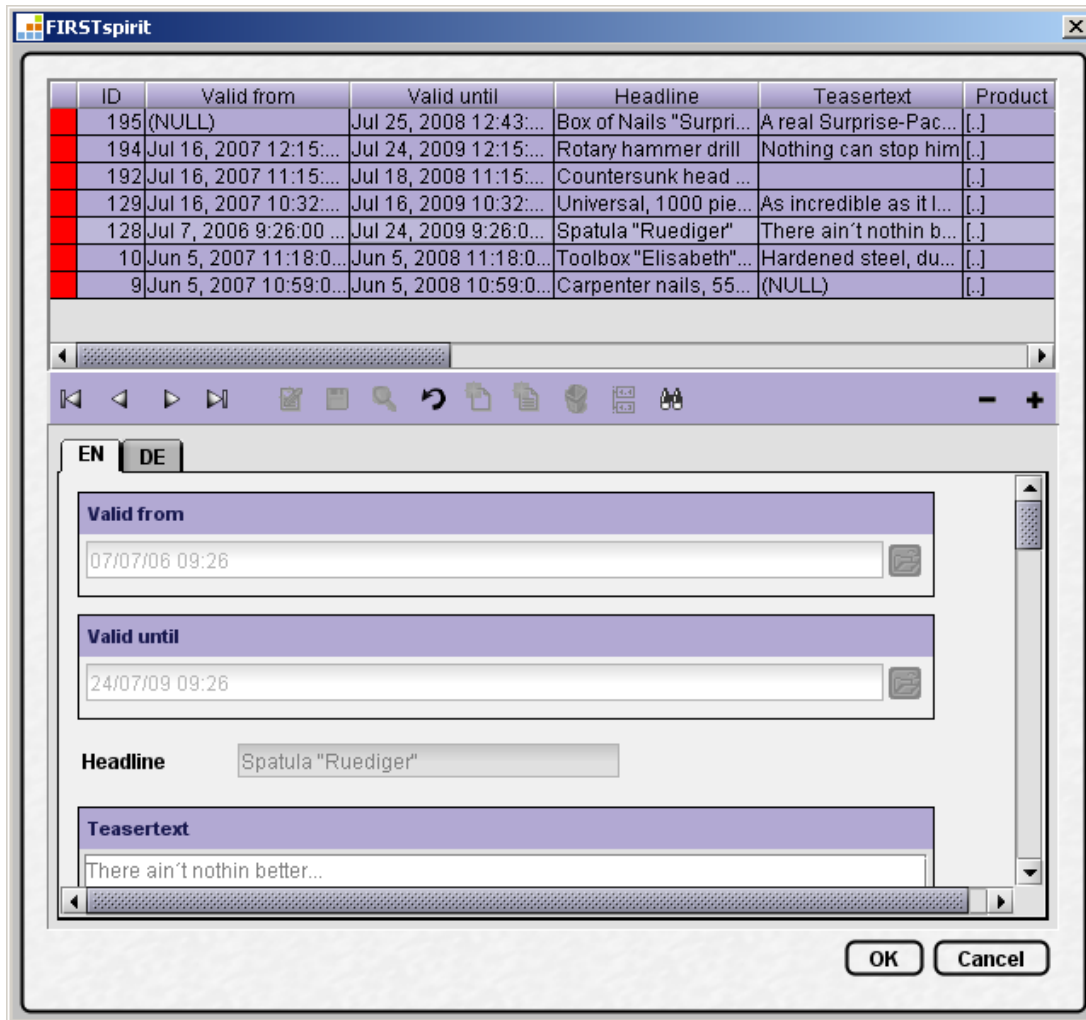
Figure 10-23: Add data record

The icons for editing a table are disabled in this window. Only the icons for convenient selection of a data record are available. A detailed description of the individual functions is given in Chapter 5.4.2 page 150.

> ⚠ *From FirstSpirit Version 4.2 Release 2 the displayed data sets can be limited to a specific number (see Chapter 5.4.1.2 page 149).*

🗑 Delete Data Record; click this icon to delete the selected data record from the list.

△ Move Data Record Up; click this icon to move the selected data record up in the list by one position.

▽ Move Data Record Down; click this icon to move the selected data record down in

the list by one position.

> ! This input component has been extended for the use of media galleries in FirstSpirit version 4.2 (see Chapter 11.9 page 370).

10.7 Select/Edit Data Record (CMS_INPUT_OBJECTCHOOSER)

Figure 10-24: Input component – Select/Edit data record

> ! This input component has been extended in FirstSpirit version 4.2. The buttons decribed below has been replaced by new icons. For information about the new functions from version 4.2 and the new layout see Chapter 10.7.1 page 268.

This input component can be used to select an individual data record from a data source. The selected data record is only a reference to the respective object in the Content Store. In addition, individual data records can be changed and new data records added to enable fast access to the data of the linked data source.

Reset Click this button to remove the selected data record from the input

components.

New Click this button to open a window for entering a new data record.
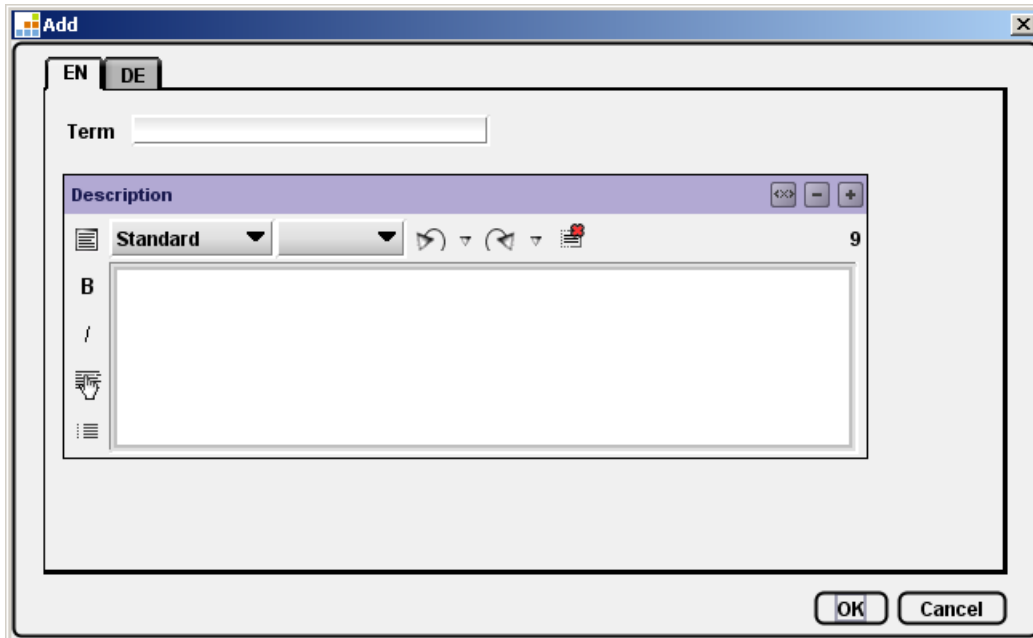
Figure 10-25: Add data record

Contents can now be inserted for all input components of the data record. Click the **OK** button to add the data record to the database table.

Select Click this button to open a window for selecting a data record from the integrated database table.
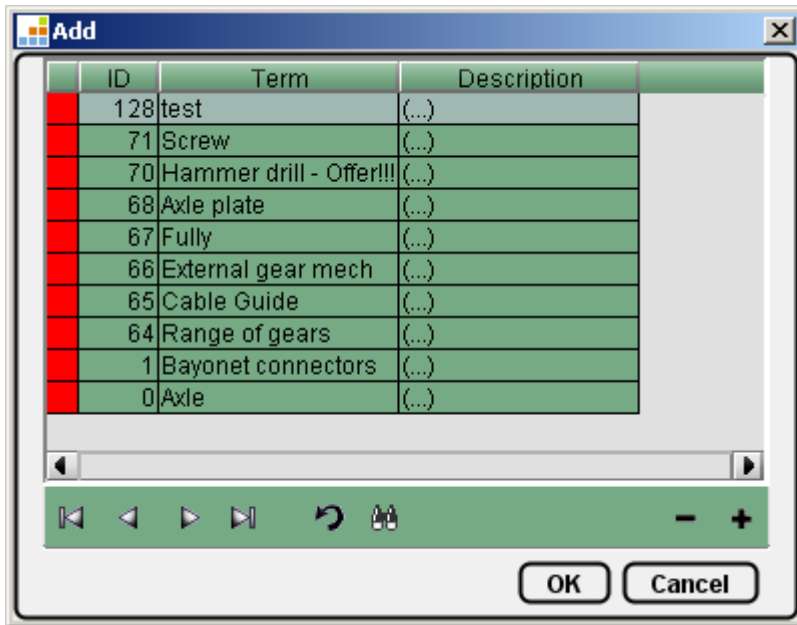
Figure 10-26: Select data record

Icons for convenient selection of a data record are available in the bottom bar. A detailed description of the individual functions is given in Chapter 5.4.2 page 150.

> **!** *From FirstSpirit Version 4.2 Release 2 the displayed data sets can be limited to a specific number (see Chapter 5.4.1.2 page 149).*

Edit Click this button to open a window for editing the selected data record.
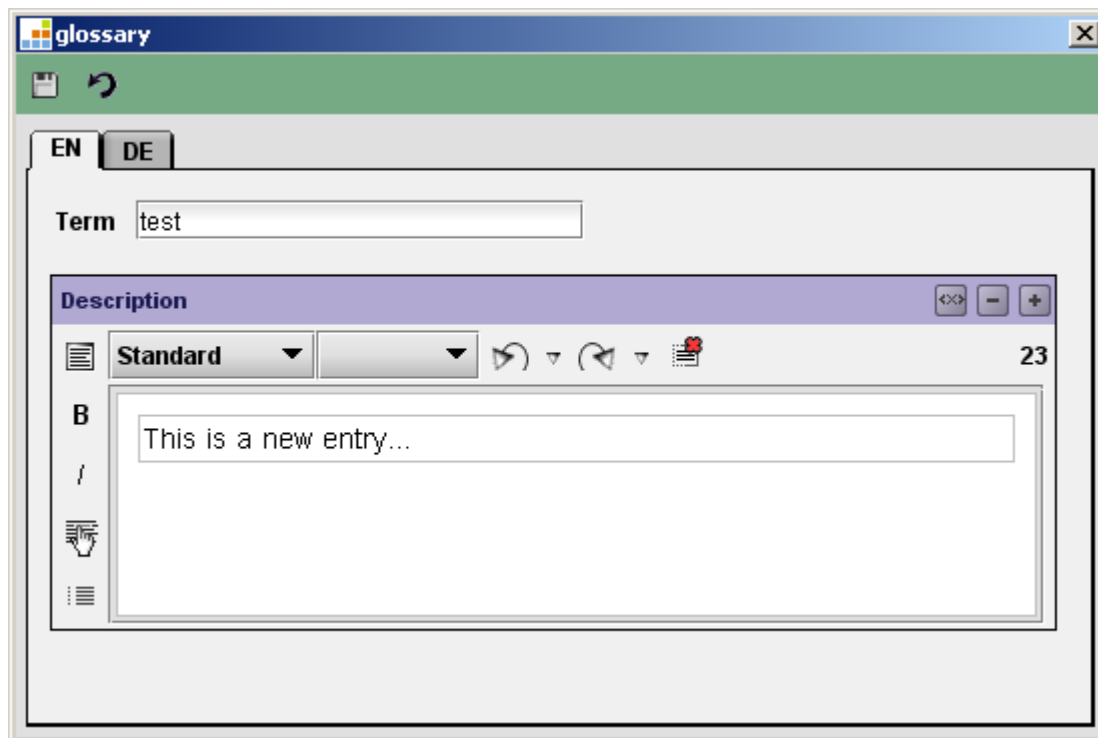
Figure 10-27: Edit data record

After the required contents of the data record have been changed the changes can be saved by clicking the ▣ icon. Click the ↺ icon to undo the changes made to an existing data record.

10.7.1 Enhancements in FirstSpirit Version 4.2

With FirstSpirit Version 4.2, the input component has been enhanced and the layout revised. 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.

Object Chooser

The dataset is also used at another place. | Show usages

Salutation

Ms

Firstname

Petra

Lastname

Presse

Phone-Nr

eMail

Figure 10-28: Object chooser input component from FirstSpirit Version 4.2

The elements of the input component are displayed with the colour concept of the respective Store, in Figure 10-28 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.

Select: the selection dialog for database content is used to select the data set
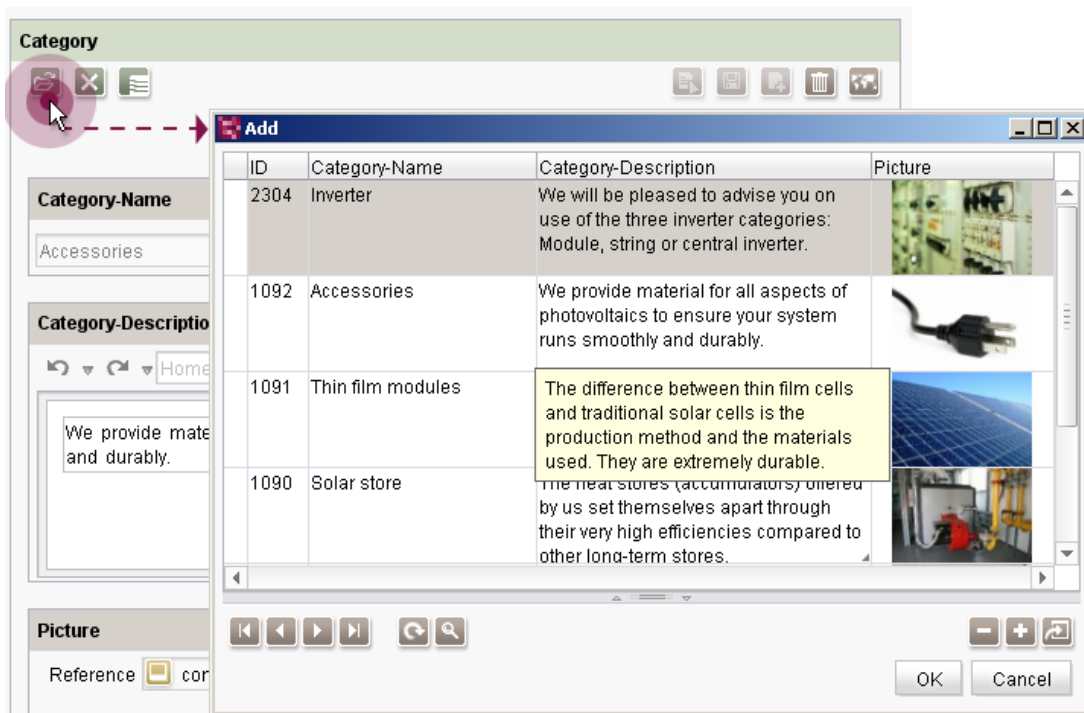
Figure 10-29: Object chooser input component with selection dialog

Jump to; click this icon to switch to the Content Store in order to edit the selected data record there. If a filter or search is active on the content (data source) concerned (see chapter 5.4.1.2 page 149 and chapter 5.4.2.1 page 151 ff.), it might not be possible to display the selected data record.

Delete: depending on the user's rights/permissions and the template developer's predefined settings, this icon can not only be used to remove the dataset from the input component ("Reset"), but also to delete it from the Content Store.

Display language tabs: click this icon to show or hide the language dependent database content via tabs within the input component.

Show usages If the dataset is referenced elsewhere, the number of uses is visualised in the input component. Click this button to display the other uses.

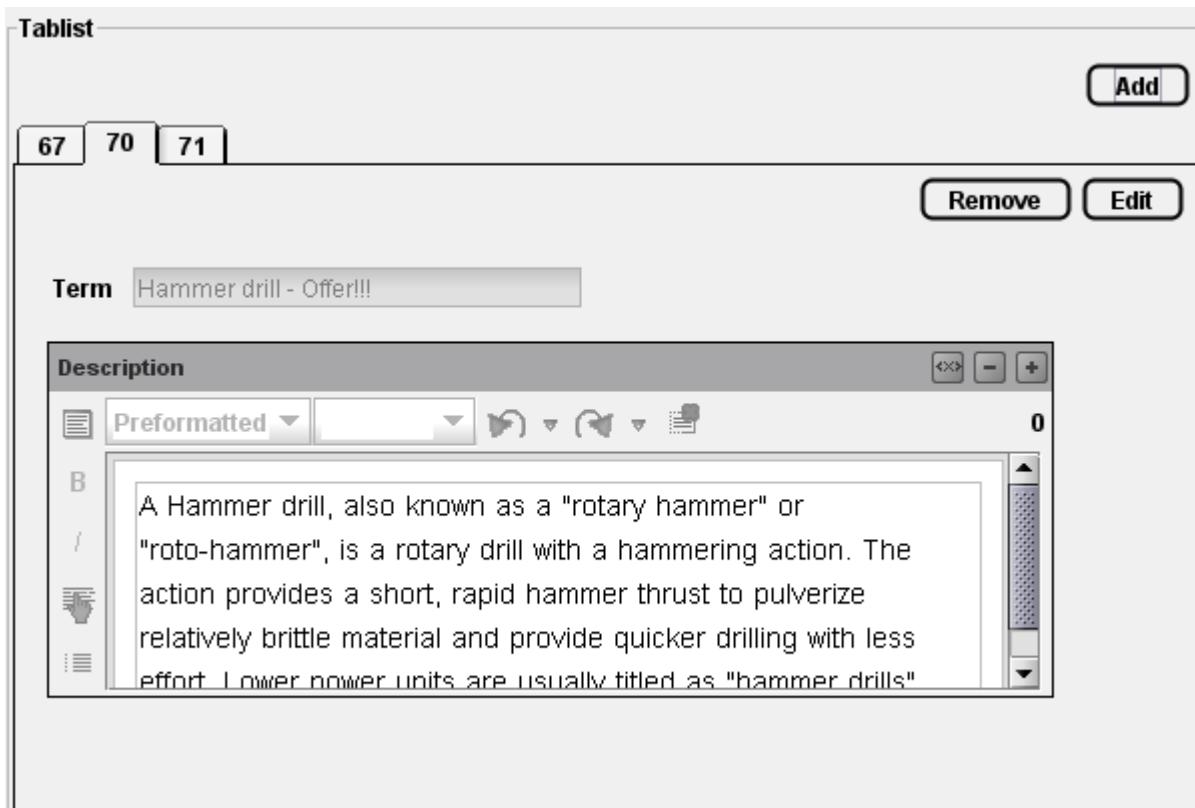## 10.8 Data Record Tab List (CMS_INPUT_TABLIST)

Figure 10-30: Input component – Tab list

This input component can be used to access the data records of a linked database table. The individual tabs are labelled via a reference table attribute. The individual input fields can be edited within the tabs.

> *This input component is only available within the Content Store.*

Add Click this button to copy a new data record from the reference table into the tab list.

Remove Click this button to remove the selected data record from the tab list. However, this data record continues to exist in the reference table.

Edit Click this button to edit the selected data record. The changes are also stored in the reference table.

10.9   Date and/or Time Selection (CMS_INPUT_DATE)

CMS_INPUT_DATE   `11 01 2007`

Figure 10-31: Input component – Date and/or time selection

The date and/or time selection can be used to select a date and time, from FirstSpirit Version 4.2R2 date and/or time. Click the ⬚ icon to open a window in which the date and/or time can be selected (see Figure 10-32). Click the ✖ icon to remove the selected date and/or time from the input component.

If the project developer has defined a preset default value for the input component the ⬚ "Accept Default Value" icon is located on the right next to the component. (For further information, see Chapter 11.10 page 386).

Figure 10-32: Select date

A new date can be selected within the window which is then automatically transferred into the input component in the format which has been specified by the template developer. When the dialog window is opened the currently entered date is displayed as a default (red border). The user can click the calendar view to select a new date. Use the ◀▶ icon to page forward/back a month and use the ◀◀▶▶ icon to page the calendar view forward/backward by a year. In addition a time can be entered in the bottom part of the date selection. Regional and national public holidays are highlighted in the calendar view in different colours. A time can be entered in the lower area of this window.

From FirstSpirit Version 4.2R2 an input component for only selecting a date or a time can be displayed instead of the input component for date **and** time.

According to the configuration of the project developer date and/or time can be entered directly into the input component, namely in a preset format. If the format of the entry differs from the requested format the date/time can not be saved. An error message will be displayed instead of

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386.)

10.10 File Selection (CMS_INPUT_FILE)

**Figure 10-33: Input component – File selection**

File selection enables files from the Media Store to be referenced to a page or a section in the Page Store. Depending on the configuration and the access permissions, this input component can be used to reference files from the local Media Store and/or the Media Store of a remote project.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

Reference: A reference to the required file from the project's Media Store (or of a remote project, from FirstSpirit version 4.2 of several remote projects) is entered here. The buttons at the end of the input field are used to select a file or to delete an already existing reference. The buttons can be used as follows:

Click the icon to open a window in which the Media Store tree structure is displayed (see Figure 10-34).
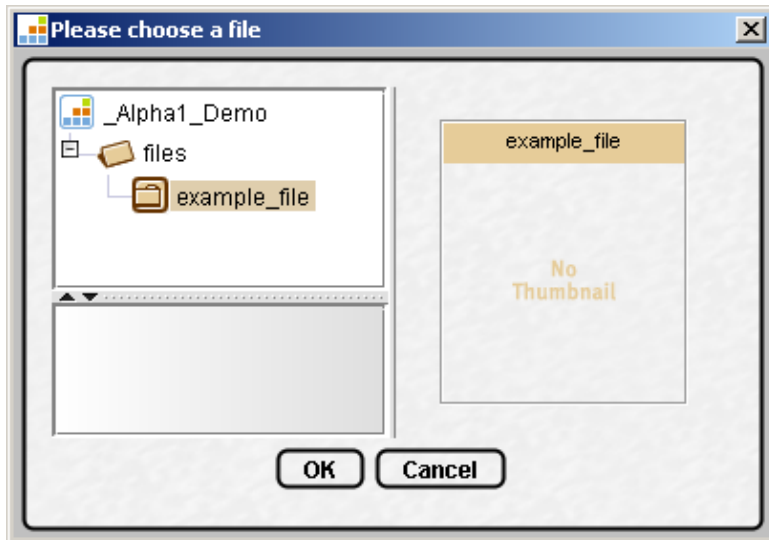
Figure 10-34: Select file

> ❗ *The media selection of this component is solely limited to all media created as a file (and not as a picture) in the Media Store.*

> ❗ *A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the look & feel "Classic" the file selection works as follows.*

The required file can be selected from the tree structure. If a medium is clicked information about the file type appears on the right-hand side to make the file selection easier for the user.

This icon is displayed depending on the configuration of the input component. If the component allows uploading of files into the project's Media Store, click this icon to open a dialog window for selecting a target folder of the Media Store. (If no folders exist in the Media Store, the root node can be selected as upload target.).

Depending on the configuration of the input component, the upload can also take place in a remote project (see Chapter 10.3.2 page 255).

- Create language-dependent medium   (see Chapter 10.3.3 page 258).
- Create language-independent medium (see Chapter 10.3.4 page 261).

Click the icon to switch to the selected medium in the Media Store.

⊠ Click the icon to remove the selected medium from the input component.

Comment: If required a comment on the file can be entered here.

The selection and creation of a file is analogous to the picture selection component (see Chapter 10.3 page 251).

The input component for file selection supports the licence-dependent "Remote Media" function.

10.10.1 Selection of media from a remote project ("Remote Media")

> ⚠ *The "Remote Media" function is a licence-dependent additional module.*

The selection of files (from the Media Store) from a remote project is analogous to the selection of pictures (from the Media Store) (see Chapter 10.3.1 page 253).

10.10.2 Uploading media into a remote project ("Remote Media")

> ⚠ *The "Remote Media" function is a licence-dependent additional module.*

The uploading of files into a remote project is analogous to the uploading of pictures (see Chapter 10.3.2 page 255).

10.10.3 Create language-dependent media

The language-dependent creation of files (in the Media Store) is analogous to language-dependent creation of pictures (in the Media Store) (see Chapter 10.3.3 page 258).

10.10.4 Create language-independent media

The language-independent creation of files (in the Media Store) is analogous to language-independent creation of pictures (in the Media Store) (see Chapter 10.3.4

page 261).

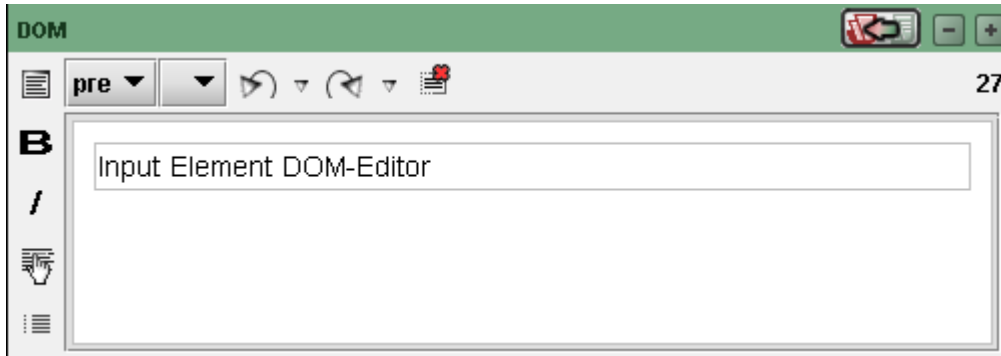## 10.11 DOM Editor (CMS_INPUT_DOM)

Figure 10-35: Input component – DOM Editor

This input component is provided for larger text inputs with simple formatting and reference inputs. The following aids which can however be limited by the definition in the Template Store are available when using this Quick Editor:

If the project developer has defined a preset default value for the input component the [icon] "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

> [!] *Project developers can make available so-called "inline tables" to the editor from FirstSpirit version 4.1. For more information see Chapter 11.6.10 page 351.*

> [!] *According to the settings of the template developer the functionality of the licence-dependent module "FirstSpirit Office" can be available from version 4.2. This module is used for the import of formatted content from word documents using the icon [icon]. For detailed information see documentation about the module "FirstSpirit Office".*

Depending on the project's configuration, a spelling check can be carried out within the input component (see Chapter 11.6.3 page 328).

[icon] Large Edit Window; click this icon to open the Editor window in full screen size. The window is closed by clicking the "x" icon in the top right-hand corner. The

contents of the large window are then automatically copied into the small window.

Standard ▼ Predefined formatting which is to apply to the whole section can be selected here. The available formatting is defined by the project developer.

Underline ▼ Predefined formatting which is to apply to the selected text only can be selected here. The available formatting is defined in the Format template and is specified by the project developer.

▼ Undo; this icon can be used to undo entries or formatting made. Click the arrow to open a list with all the most recently made changes so that several changes can be undone simultaneously.

▼ Restore; use this icon to restore entries and formatting which have been undone. Click the arrow to open a list with all the most recently undone changes so that several changes can be restored simultaneously.

Hide Lines; use this icon to hide all reference lines in the DOM Editor. Click the icon again to display the lines once more.

B Bold; this icon can be used to highlight selected text in the editor with bold text. Further, corresponding visible markers (tags) are inserted at the start and end of the bold selected text. ( )

I Italic; this icon can be used to highlight selected text in the editor in italics. Further, corresponding visible markers (tags) are inserted at the start and end of the italicised selected text. ( )

Insert Link; use this icon to insert a link in the cursor's current position. The procedure for setting a link is analogous to the description of the "Link Input" input component. (See Chapter 10.23 page 291) ( )

Insert List; use this icon to insert a list in the current section.

> ❗ *As the DOM Editor is a very complex input component, detailed documentation on working with the DOM Editor input component is provided again in Chapter 11.6 page 326.*
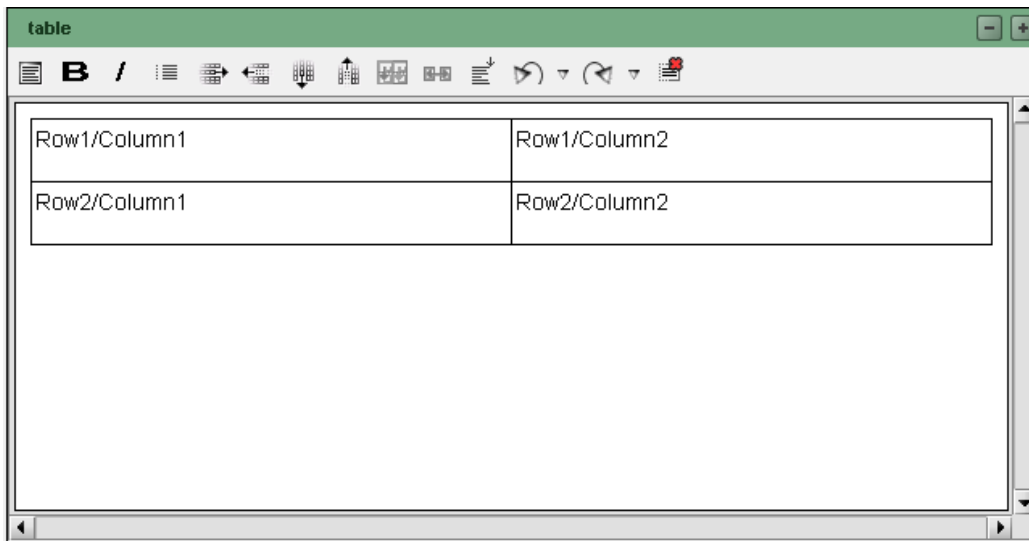
10.12 DOM Table (CMS_INPUT_DOMTABLE)

Figure 10-36: Input component – DOM Table

This input component can be used to enter larger text inputs with simple formatting and link inputs directly in the table view.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

Depending on the project's configuration, a spelling check can be carried out within the input component (see Chapter 11.6.3 page 328).

 Large Edit Window; click this icon to open the Editor window in full screen size. The window is closed by clicking the "x" icon in the top right-hand corner. The contents of the large window are then automatically copied into the small window.

Underline ▼   Predefined formatting which is to apply to the selected text only can be selected here. The available formatting is defined in the Format template and is specified by the project developer.

B Bold; this icon can be used to highlight selected text in the editor with bold text. Further, corresponding visible markers (tags) are inserted at the start and end of the bold selected text. ( )

I Italic; this icon can be used to highlight selected text in the editor in italics. Further, corresponding visible markers (tags) are inserted at the start and end of the italicised

selected text. ( )

 Insert Link; use this icon to insert a link in the cursor's current position. The procedure for setting a link is analogous to the description of the "Link Input" input component (see Chapter 10.23 page 291). ( )

 Insert List; use this icon to insert a list in the current section.

 Add Row; click this icon to add a whole **Row** to the table below the selected cell.

 Delete Row; click this icon to delete from the table the whole **Row** in which the selected cell is located.

 Add Column; click this icon to add a whole **Column** to the table on the right next to the selected cell.

 Delete Column; click this icon to delete from the table the whole **Column** in which the selected cell is located.

 Merge Cells; click this icon to merge the selected cells with each other. The content of the selected cells is retained.

 Split Cells; click this icon to split merged cells again. The content of the merged cells is retained in the top left-hand or top cell.

 Cell Properties; click this icon to open a window with the properties of the selected cell. (For a mode detailed description of the cell properties, see Chapter 11.7.4 page 359.)

 Import: click the Import icon to import text (.txt) or CSV files (.csv), whose content is transferred directly into a table. Each line of the text/CSV file is written into a table row, columns can be generated by separators. **Important:** An import deletes all the table content to date and replaces it with the imported content.

The following window opens:
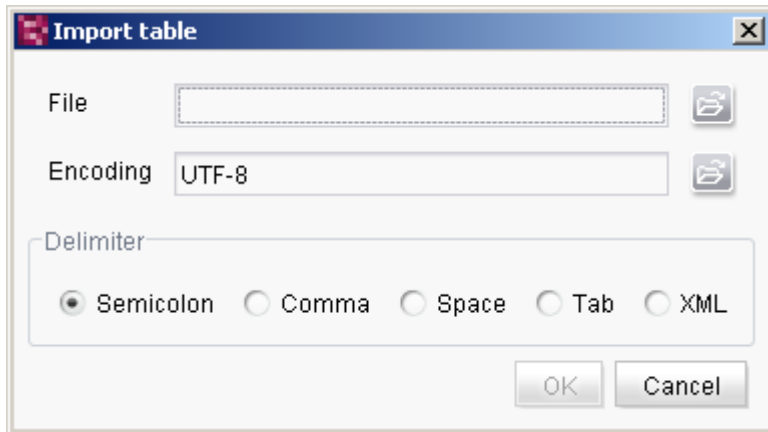
Figure 10-37: DOM Table input component: Import table

File: The ⬚ icon can be used to select the required file from the directory structure of the workstation computer.

Encoding: The ⬚ icon can be used to select the required encoding by which the source file is to be converged.

Separator: The separator which separates the content of adjacent cells can be defined here.

↶ ▽ Undo; this icon can be used to undo entries or formatting made. Click the arrow to open a list with all the most recently made changes so that several changes can be undone simultaneously.

↷ ▽ Restore; use this icon to restore entries and formatting which have been undone. Click the arrow to open a list with all the most recently undone changes so that several changes can be restored simultaneously.

⬚ Hide Lines; use this icon to hide the table frame. Click the icon again to display the frame once more.

> **!** *As the DOM Table is a very complex input component, detailed documentation on working with the DOM Table input component is provided again in Chapter 11.7 page 356.*

> ! *Additional options for formatting tables and for integrating them into the text flow are offered by the so-called "inline tables" (from version 4.1). For more information see Chapter 11.6.10 page 351.*

> ! *According to the settings of the template developer the functionality of the licence-dependent module "FirstSpirit Office" can be available from version 4.2. This module is used for the import of formatted content from word documents using the icon ⬚. For detailed information see documentation about the module "FirstSpirit Office".*

10.13 Image Map (CMS_INPUT_IMAGEMAP)
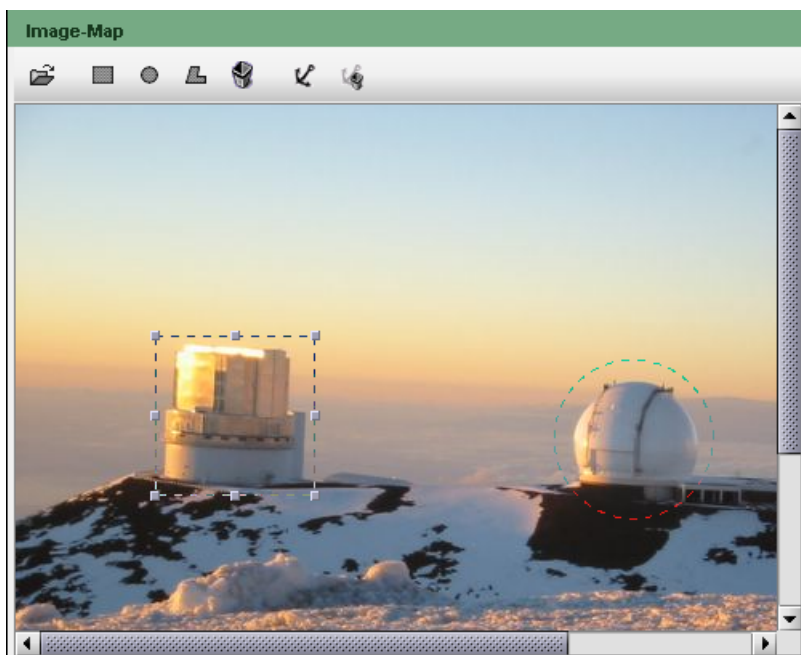
Figure 10-38: Input component – Image Map

This input component can be used to integrate references (links) to various positions of a selected background picture. A framed object is first generated for each link and the shape and size is adapted to the required position of the background image. The reference to the framed object can then be integrated.

If the project developer has defined a preset default value for the input component the ⬚ "Accept Default Value" icon is located in the top right-hand corner. (For

further information, see Chapter 11.10 page 386).

Select Background Picture; click this icon to open a window in which the tree structure of the Media Store of the local project and/or of a remote project (from FirstSpirit version 4.2 of several remote projects) is displayed.
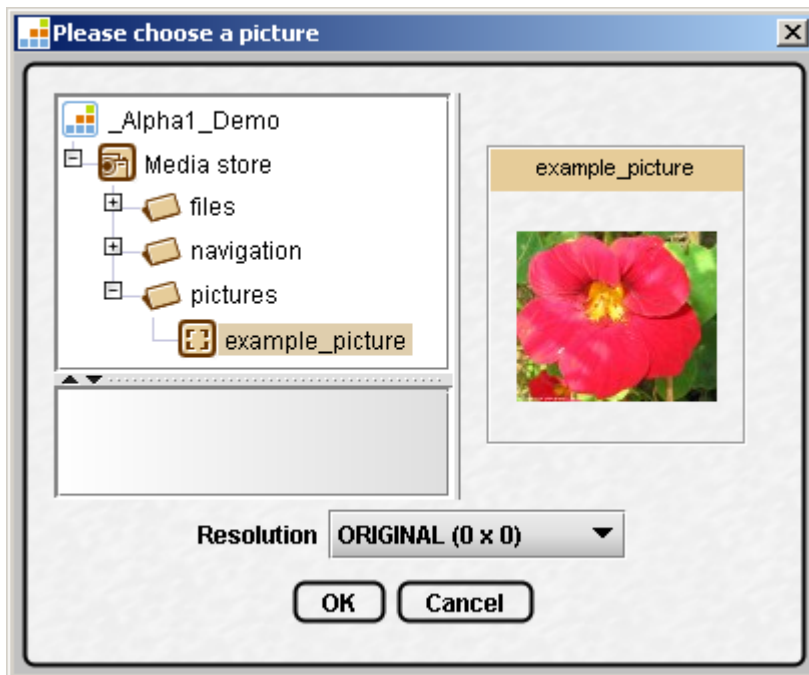
Figure 10-39: Select background picture

A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the Look & Feel "Classic" the picture selection works as follows.

The background picture can be selected from the tree structure. If a medium is clicked a thumbnail appears on the right-hand side to make the picture selection easier for the user. Which of the available resolutions is to be used for the background picture must also be specified in the **Resolution** field.

Generate rectangle; click this icon to generate a frame object in the form of a rectangle on the background picture. The size and position of the rectangle must then be adjusted again manually.

Generate circle; click this icon to generate a frame object in the form of a circle on the background picture. The size and position of the circle must then be adjusted

again manually.

Generate polygon; click this icon to generate a frame object in the form of a polygon. Each of the individual points of the polygon can then be specified by clicking the background picture with the mouse.

Delete Object; click this icon to remove the selected frame object.

> *Before deleting the system does not check whether a reference to the frame object is integrated.*

Insert/Edit Link; of a link does not yet exist for the selected frame object, click the icon to insert a new link. The procedure for setting a link is analogous to the description of the "Link Input" input component (see Chapter 10.23 page 291). If a link already exists the input window of the type of link used opens.
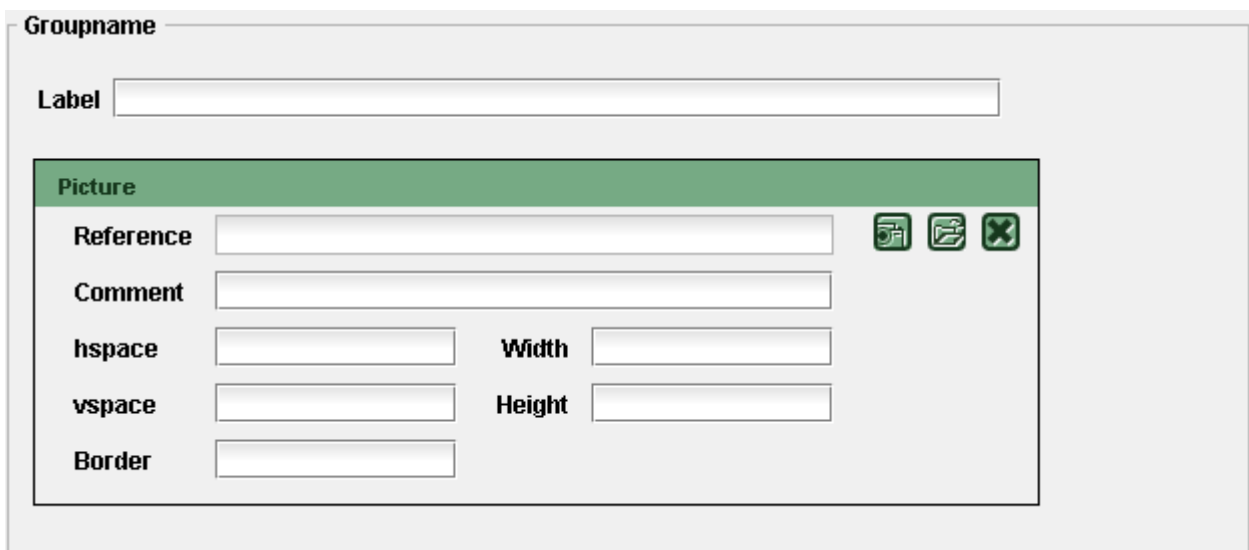
Delete link; click this icon to delete the integrated link from the selected frame object.

10.14 Component Grouping (CMS_GROUP)

This component is a special case as it is not used for including editorial contents but for the graphical grouping of further input components. The required input components can be collated together to form a group.

10.14.1 Display a grouping with frame

Figure 10-40: Grouping input component (frame display)

For the display with frame all the corresponding input components are identified by a common frame. Each group has a common group name.

In Figure 10-40 the single line text input (see Chapter 10.20 page 289) and picture selection (see Chapter 10.2 page 249) input components were gathered together to form a group.

10.14.2 Display a grouping with tabs
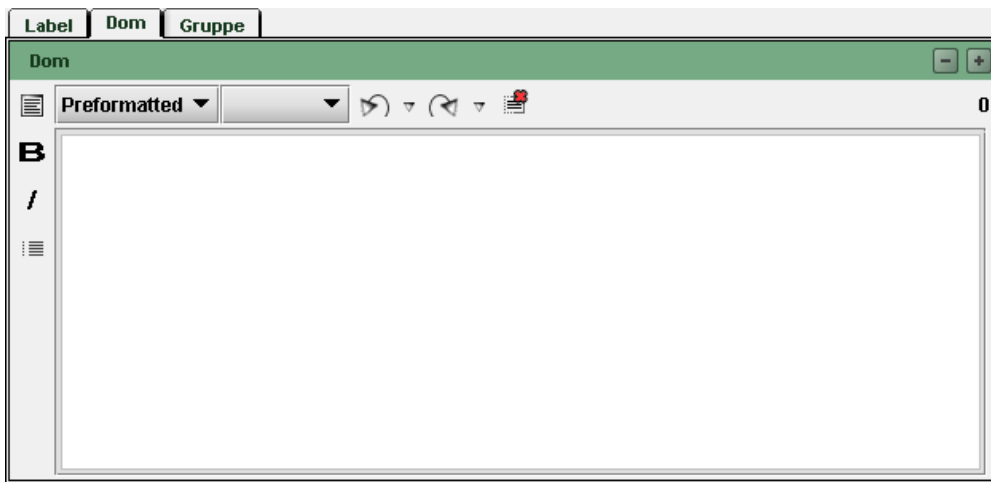
Figure 10-41: Grouping input component (tab display)

For the display in tabs the corresponding input components are displayed in various tabs.

In Figure 10-41 in the first tab the input component multi-row text input (see Chapter 10.21 page 290) and in the second tab the DOM Editor component (see Chapter 10.11 page 276) was used. In the third tab a new group of input components was put together.

10.15 List (CMS_INPUT_LIST)
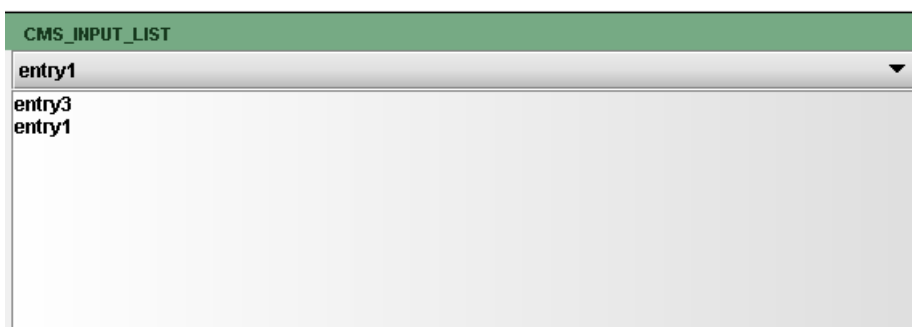
Figure 10-42: Input component – List

The list input component provides the user with an option for selecting inputs defined by the project developer in the Template Store. The input is selected by clicking the arrow symbol behind the input row. The selection list then opens, from which the required input can be selected. It is possible to simultaneously make more (or less)

than one selection for the list.

If the project developer has defined a preset default value for the input component the [icon] "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

10.16 Number (CMS_INPUT_NUMBER)

CMS_INPUT_NUMBER 2

Figure 10-43: Input component – Number

This input component is provided for the input of numerical values. The value range and the type of numbers which can be entered (integer, decimal number, ...) are specified just like the display format for the numbers by the project developer in the Template Store.

10.17 Radio Buttons (CMS_INPUT_RADIOBUTTONS)

CMS_INPUT_RADIOBUTTON
- ○ entry4
- ● entry3
- ○ entry2
- ○ entry1
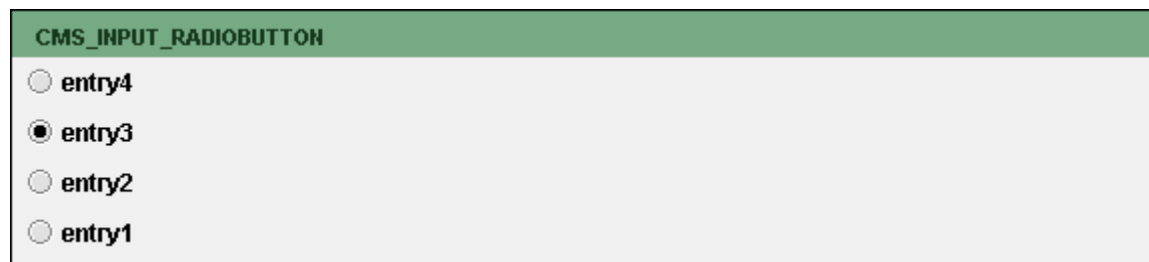
Figure 10-44: Input component – Radio Buttons

Radio buttons provide the user with an option for selecting inputs defined by the project developer in the Template Store. Precisely one selection is always possible here.

If the project developer has defined a preset default value for the input component the [icon] "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

10.18 Permissions Definition (CMS_INPUT_PERMISSION)
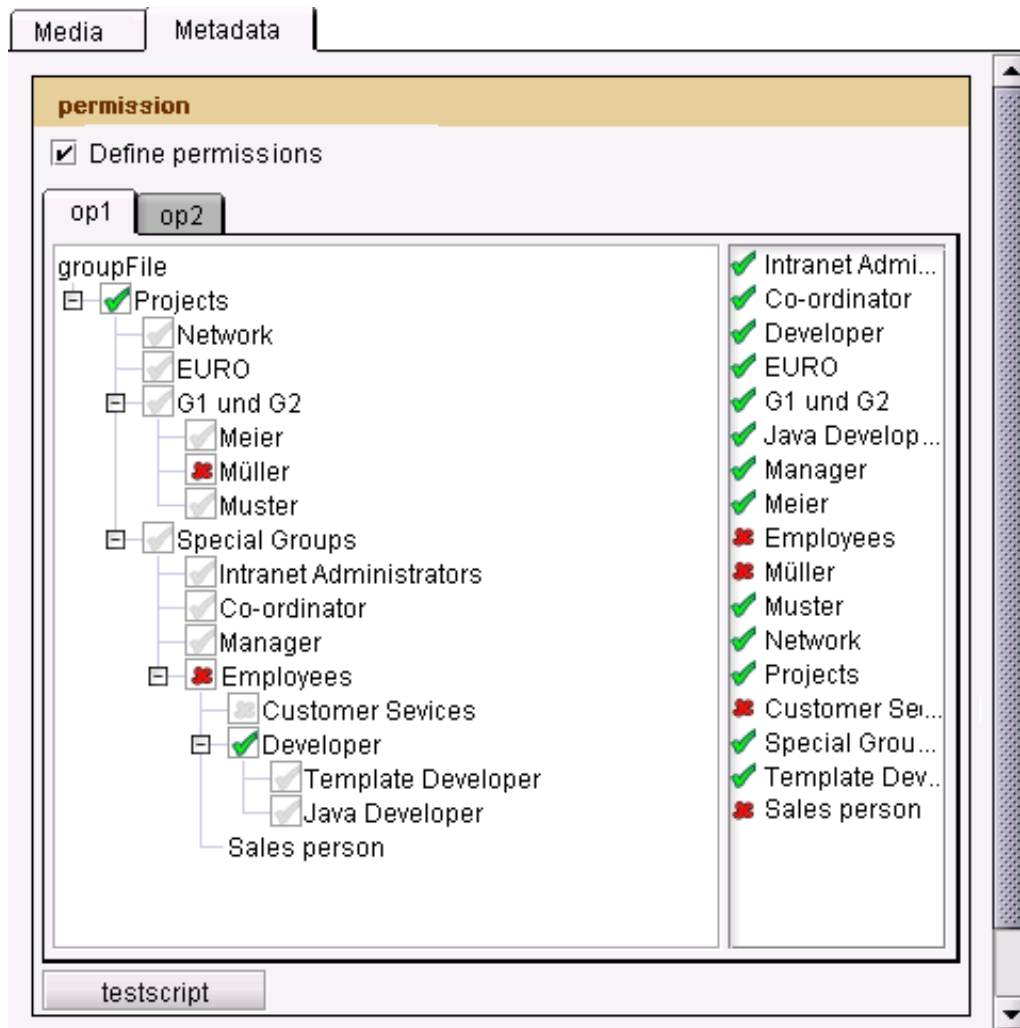
Figure 10-45: Input component – Permissions definition

This input component can be used to assign user permissions on the basis of a hierarchical group definition. This input component is usually used within the scope of the metadata definition.

> ! *As the permissions definition is a very complex input component, detailed documentation on working with the user permissions input component is provided again in Chapter 13.3.2 page 472.*

10.19 Page Reference (CMS_INPUT_PAGEREF)
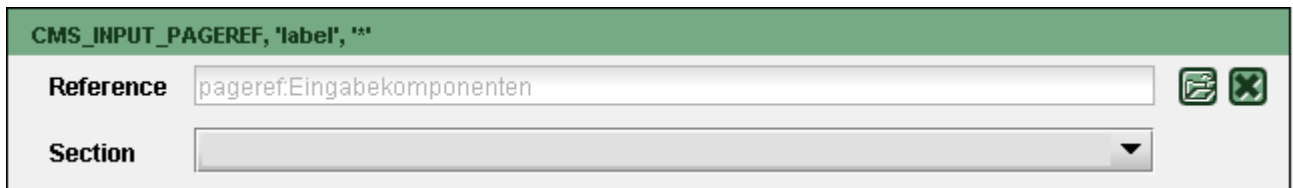
Figure 10-46: Input component – Page reference

This input component is used to integrate references to pages of the Site Store to a page or a section in the Page Store.

If the project developer has defined a preset default value for the input component the [icon] "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

Reference: Click the [icon] icon to select a page from the Site Store. A window opens in which the tree structure of the Site Store is displayed.

Figure 10-47: Select page reference

> **!**   *A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the Look & Feel "Classic" the page selection works as follows.*

The required page can be selected from the tree structure.

> ❗ *From FirstSpirit version 4.2 on, the Site Stores of one or more remote projects can be displayed here to select the required page.*

Click the ⊠ icon to remove the selected medium from the input component.

Section: If pages have several sections, select which of these sections are to be referenced.

> ❗ *Unlike the input of a link in text input components, here it is not necessary to specify the text to be highlighted on the website. This already took place in the template.*

> ❗ *A variation of this input also has a further "Comment" row. The text which can be entered here appears later on the displayed page if the mouse is moved over the reference.*

10.20 Single-Row Text Input (CMS_INPUT_TEXT)

CMS_INPUT_TEXT | text input | 🔲

Figure 10-48: Input component – Single row text input

This input component is provided for single row text inputs, e.g. headings. Formatting (bold, italics, etc.) cannot be selected here, they are uniformly specified via the template development.

If the project developer has defined a preset default value for the input component the 🔲 "Accept Default Value" icon is located on the right next to the component. (For further information, see Chapter 11.10 page 386).

10.21 Multi-Row Text Input (CMS_INPUT_TEXTAREA)

CMS_INPUT_TEXTAREA, 'label', '*'

text input multiline

Figure 10-49: Input component – Multi-row text input

This input component has been conceived for larger text inputs which are not to be formatted.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

10.22 Toggle (CMS_INPUT_TOGGLE)
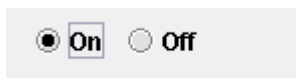
On   Off

Figure 10-50: Input component – Toggle

This input component can be used to toggle between two specified values (e.g. on/off, right/left).

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located on the right next to the component. (For further information, see Chapter 11.10 page 386).

10.23 Link Input (CMS_INPUT_LINK)
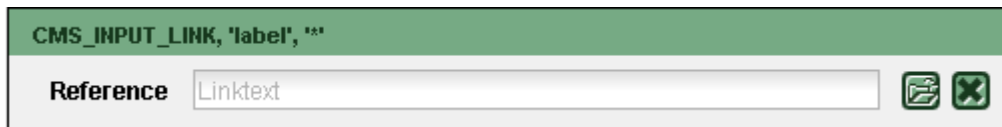
Figure 10-51: Input component – Link input

Reference: Click the ⬚ icon to create a link. A window opens in which the link type must be selected.
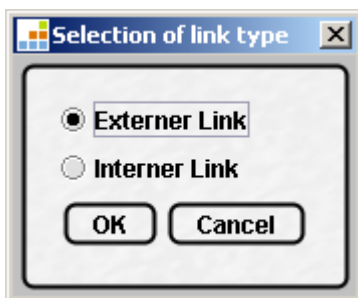
Figure 10-52: Link type selection

After selecting the link type an input window is displayed whose content depends on the link type.

If the project developer has defined a preset default value for the input component the ⬚ "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

> ❗ *The project developer can use the Template Store to fill the individual fields with content in advance. However, these default values can be overwritten with new input.*

> ▣ *The project developer determines which of the respective input fields are used for the link. It is therefore possible that individual inputs do not have any effect or errors occur in the display if inputs are missing.*

10.23.1 Internal Link

Figure 10-53: Internal Link

> ▣ *From FirstSpirit version 4.2 internal links can be created on the basis of "Generic link editors", too. User guidance and layout will then differ from the representation below (see Chapter 10.23.5 page 303).*

This type of link is a link to another page in the same FirstSpirit project.

Target: The anchor symbol ↙ can used to select a page reference to be linked to from the Site Store.
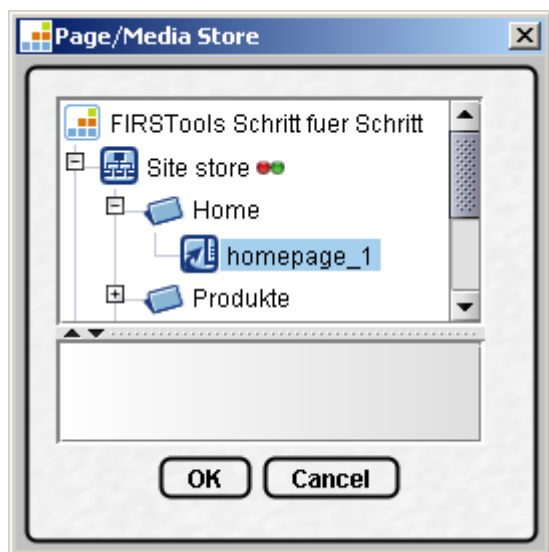
Figure 10-54: Select page reference

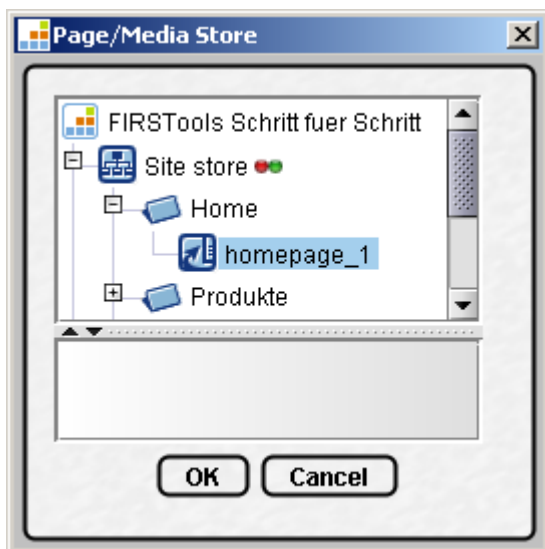> ! *A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the Look & Feel "Classic" the picture selection works as follows..*

The link can also be entered manually. If entering manually it is necessary to ensure that the link syntax is correct (pageref: *Reference name*) as otherwise a broken link will result.

Section: After selecting a target it is possible to use a selection list to select a section from the target page to more precisely define the link.

Link Text: The link text to be displayed on the website can be entered in this field.

Comment: A note for the visitor to the website can be entered here; they then see the note when they move their mouse over the link (Mouse-over effect).

Link template: Use this field to select which link template is to be used for creating the link.

Image: The anchor symbol ⤴ can used to select a medium to be linked to from the Medium Store.
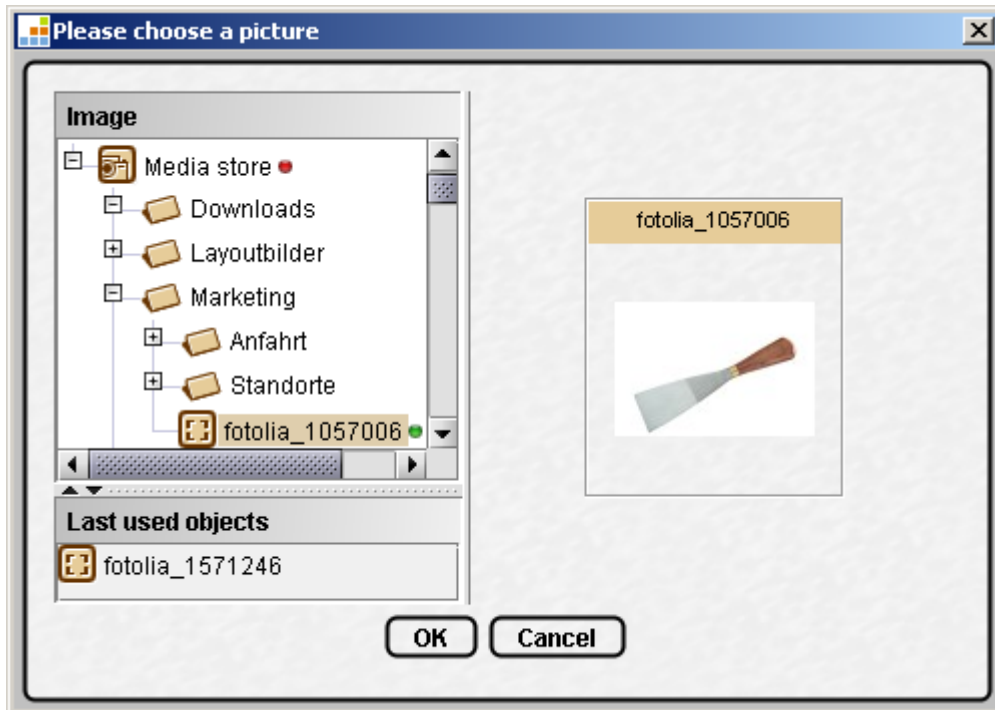
Figure 10-55: Select Medium

> ❗ *A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the Look & Feel "Classic" the picture selection works as follows.*

Frame: This input field can be used to specify a target frame or a target window for the link. If the project developer evaluates this field they will usually pre-assign a list to it so that a target can be selected.

Language: This field can be used to select to which project language the selected page or selected medium is to link. If nothing is entered here the currently edited language is retained. If the content is not available in this language the program automatically attempts to link to the project's master language.

10.23.2 External Link
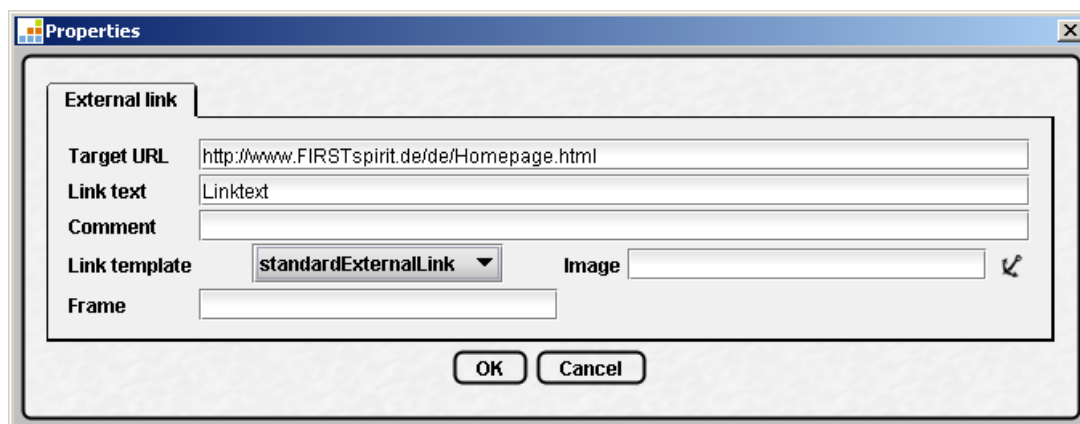
Figure 10-56: External Link

> ❗ *From FirstSpirit version 4.2 external links can be created on the basis of "Generic link editors", too. User guidance and layout will then differ from the representation below (see Chapter 10.23.5 page 303).*

This type of link is a link to a page of a public website. The following information can be entered in this input window:

Target URL: The complete URL of the target page is entered here. (e.g. http://www.FirstSpirit.de/de/Homepage.html)

Link Text: The link text to be displayed on the website can be entered in this field.

Comment: A note for the visitor to the website can be entered here; they then see the note when they move their mouse over the link (Mouse-over effect).

Link template: Use this field to select which link template is to be used for creating the link.

Image: The anchor symbol ⤦ can used to select a medium to be linked to from the Medium Store. (see Figure 10-55: Select Medium on page 294).

Frame: This input field can be used to specify a target frame or a target window for the link. If the project developer evaluates this field they will usually pre-assign a list to it so that a target can be selected.
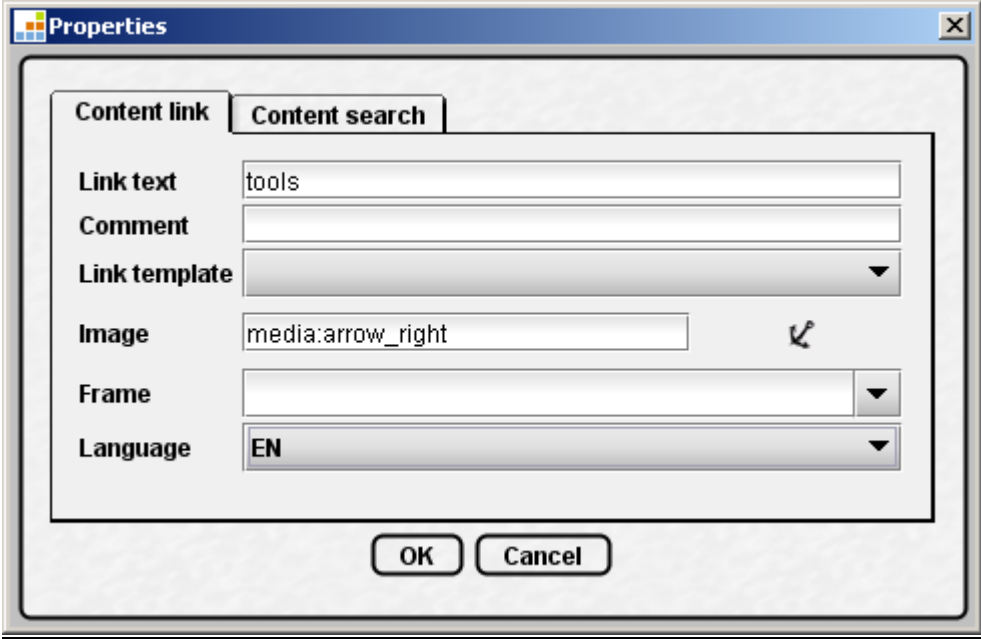
10.23.3 Content Link

This type of link is a link to a data record from the Content Store of the FirstSpirit project.

The following information can be entered in this input window:

- Content Link tab   (see Chapter 10.23.3.1 page 296).
- Content Search tab   (see Chapter 10.23.3.2 page 297).

10.23.3.1 Content Link tab

Figure 10-57: Link to a database entry (Part 1)
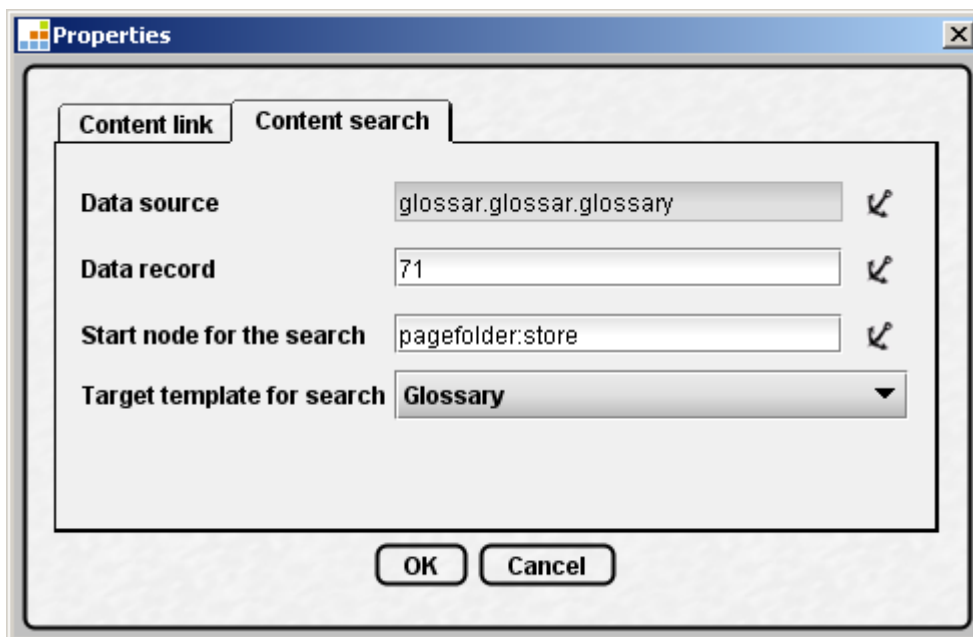
> !   *From FirstSpirit version 4.2 content links can be created on the basis of "Generic link editors", too. User guidance and layout will then differ from the representation below (see Chapter 10.23.5 page 303).*

The fields are identical to those of the **Internal Link** (see Chapter 10.23.1 Page 292).

10.23.3.2 Content Search tab

Unlike the internal or external link the Content Link is not a "direct" link to the required object. I.e. the link here is not to a data record from the Content Store but instead to a display or representation of this data record within the project. I.e., after selecting a data record ("Term" field), a data source section referenced here from the Page Store and which precisely represents the required data record is searched for within the Site Store ("Starting point of the search"). The following details must be entered in the "Content Search" tab to ensure this search is successful:

Figure 10-58: Content Search tab

!   *From FirstSpirit version 4.2 content links can be created on the basis of "Generic link editors", too. User guidance and layout will then differ from the representation below (see Chapter 10.23.5 page 303).*

Data source: One of the data sources used in this project can be selected here. Click the ⬐ icon to open the Content Store of the project to select a database table (see Figure 10-59). The selection is then copied into the Data Source field (see Figure 10-58).
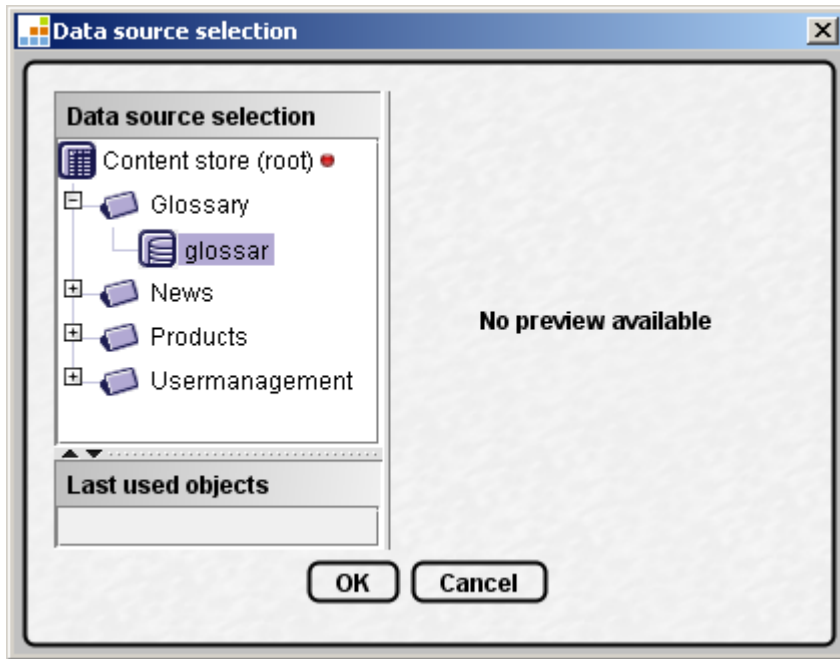
Figure 10-59: Data source selection for a content link

> *A search dialog for objects is available via this function from FirstSpirit version 4.2 (see Chapter 11.8 page 360). Up to FirstSpirit version 4.1 the data source selection works as follows.*

Data record: Click the ✓ icon to open a window with the table view of the selected data source. A data record for the link can be selected here (see Figure 10-60). Click the "OK" button to copy a reference to the data record in the data record field (see Figure 10-58).
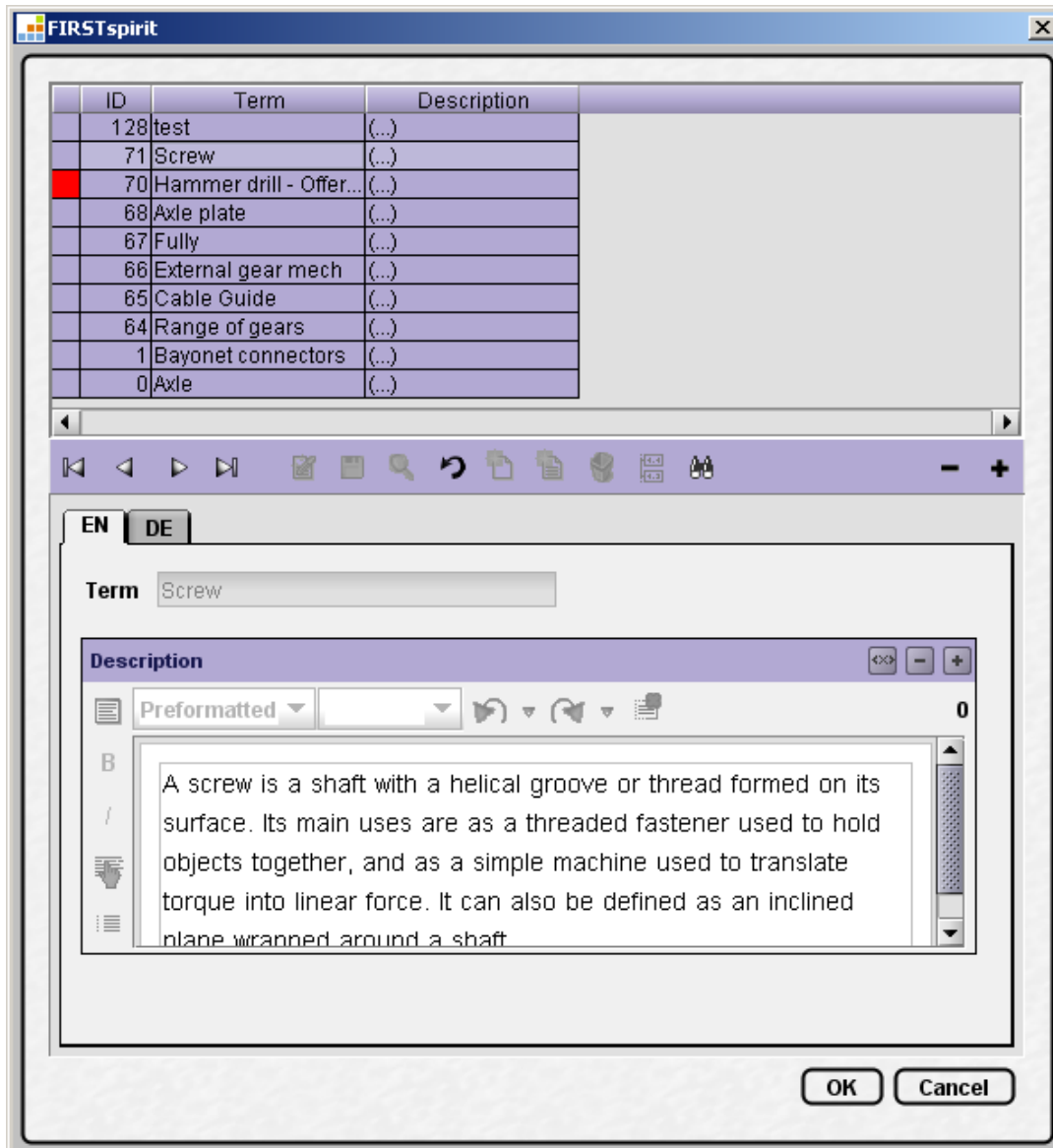
Figure 10-60: Select data record

Start node for the search: Click the ⬢ icon to open a window with project's Site Store. A page or folder from the Site Store to be used as the start node for the search for the given data record within the project can be selected here (see Figure 10-61). Click the "OK" button to copy the reference into the "Start Node for the Search" field (see Figure 10-58).
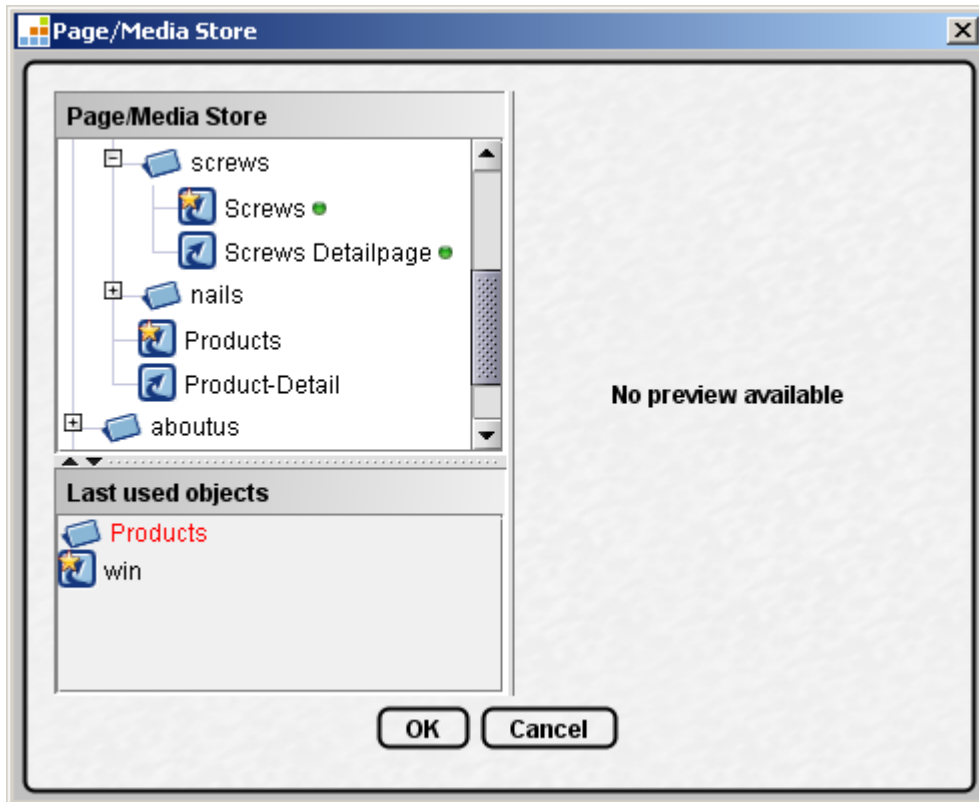
Figure 10-61: Site Store for selection of the start node for the search

> ❗ *A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the Look & Feel "Classic" the picture selection works as follows.*

Target template for search: A constraint or limit for the data record search can be entered here. A table template can be selected here to exclude search results which do not contain the required representation of the data record. This ensures that the referenced data record is displayed in the required format (e.g. in a detailed display).

10.23.4 Link to a related project

FirstSpirit supports remote access to other FirstSpirit projects. The remote access can be used within the current project to reference and display an element from the Site Store and/or Media Store of another FirstSpirit project (link to a related FirstSpirit project). The objects physically remain in the remote project.

Remote access is possible for specific, specially configured input components only. If the link template has been appropriately configured by the template developer,

selection of remote objects is supported for the link types: "internal link" (see Chapter 10.23.1) and "external link" (see Chapter 10.23.2).

Selection of a link target ("sitestoreref" attribute): When the ⬚ icon is clicked to select a link target ("sitestoreref" attribute) not only the Site Store or Media Store of the current project opens in this case but also, depending on the configuration, the Site Stores or Media Store of another FirstSpirit project.

Depending on the configuration of the input component:

- the project's local Site Store only is displayed (selection of page references to a related project is not possible).

- the project's local Media Store only is displayed (selection of media references to a related project is not possible).

- the remote project's Site Store only is displayed (selection of page references to a related project only is possible).

- the remote project's Media Store only is displayed (selection of media references to a related project only is possible).

- Both Site Stores (remote and local) are displayed (see Figure 10-62). In this case page references can be referenced from two different Site Stores. Switch over by clicking the tab of the respective project.

- Both Media Stores (remote and local) are displayed (see Figure 10-62). In this case media references can be referenced from two different Media Stores. Switch over by clicking the tab of the respective project.

All the possible combinations of this display can be set via the configuration of the link template.

Figure 10-62 shows the options for selecting remote objects (example: "internal link") as the target of a link ("sitestoreref" attribute). The remote projects configured within the link template are displayed in separate tabs in the following selection dialog. The tabs in the top part display both the Site Store or Media store of the current project ("FIRSTools_Internet (local)") and the Site Store or Media Store of the "FIRSTools" remote project.

Figure 10-62: Selection of a graphic from the remote project

> ❗ *For remote access to FirstSpirit projects the relevant remote projects must be configured beforehand by the project administrator.*
>
> ❗ *If the medium or contents have been selected via the Media Store or Site Store of the remote project they exist in the target project via this reference only, i.e. they cannot be found in the Media Store or Site Store of the target project.*
>
> ❗ *A search dialog for objects is available via this function in the Look & Feel "LightGray" from FirstSpirit version 4.1 (see Chapter 11.8 page 360). In the Look & Feel "Classic" the picture selection works as follows.*

Special "Remote Media" case: Apart from the option of selecting a link target ("sitestoreref" attribute) it is also possible to select a media object ("mediaref" attribute), which is linked e.g. instead of a text (cf. field "Link Text" in Figure 10-53).

This media object can also be selected from the Media Store of the remote project, if a project of the remote type "Remote Media" is available and the input component has been configured for selection of remote media. Depending on the configuration of the input component:

- the project's local Media Store only is displayed (selection of remote media is not possible).

- the remote project's Media Store only is displayed (selection of remote media only is possible).

- Both Media Stores (remote and local) are displayed (see Figure 10-62). In this case media can be referenced from two different Media Stores. Switch over by clicking the tab of the respective project.

> ❗ *The "Remote Media" function is a licence-dependent additional module.*

10.23.5 Generic link editors (from V4.2)

From FirstSpirit Version 4.2, input components for links can be created analogous to input components for pages and sections, by template developers by means of so-

called "Generic link editors". Thus, for entering a link all FirstSpirit input components which are described in this chapter 10 can be used. This means also, that the layout of the link input component depends heavily on the used input components, e.g. CMS_INPUT_TEXT for link texts and comments, CMS_INPUT_PICTURE or FS_REFERENCE for link images and CMS_INPUT_PAGEREF or FS_REFERENCE for internal links.

The following figure shows a comparison of the static link editors (Chapter 10.23.1 page 292 to 10.23.4 page 300) and a corresponding representation via generic editors:
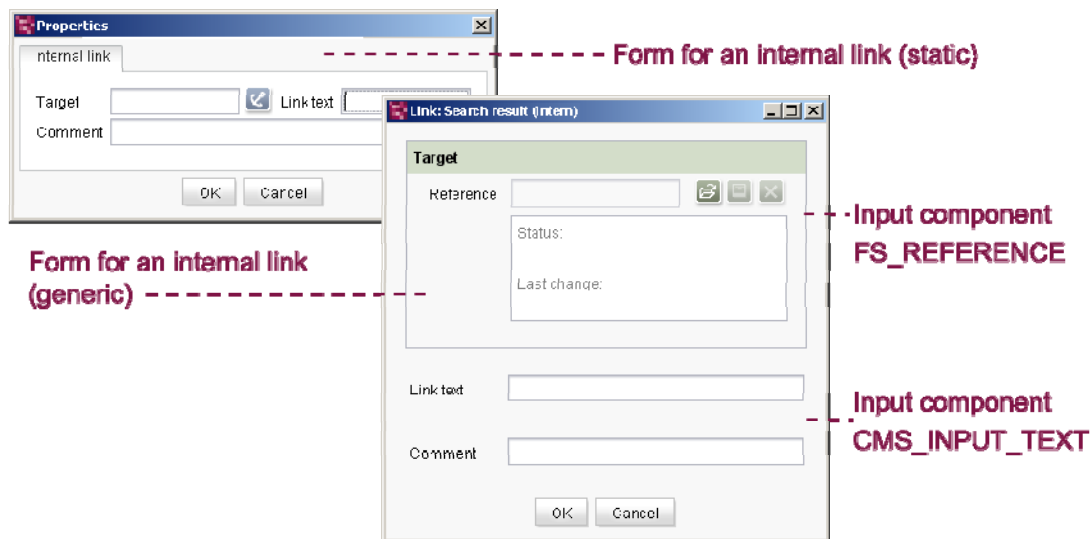
Figure 10-63: Input mask of an internal link "static" and "generic"

10.24 Link List (CMS_INPUT_LINKLIST)
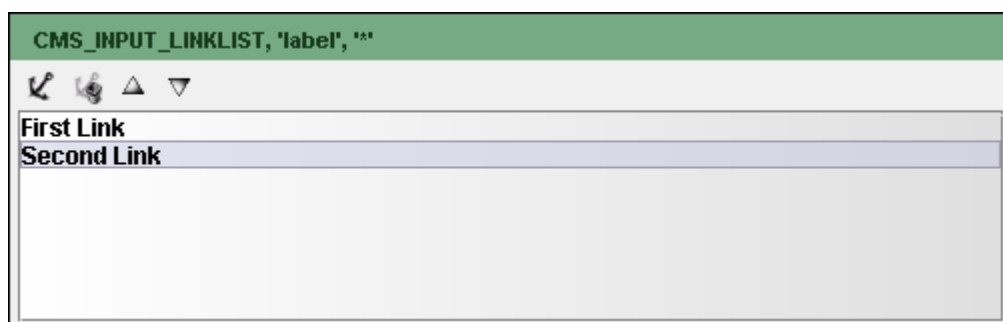
Figure 10-64: Input component – Link list

The Link List is an input component with which a list of links can be created. Four buttons are required to manage this list:

If the project developer has defined a preset default value for the input component the [icon] "Accept Default Value" icon is located in the top right-hand corner. (For

further information, see Chapter 11.10 page 386).
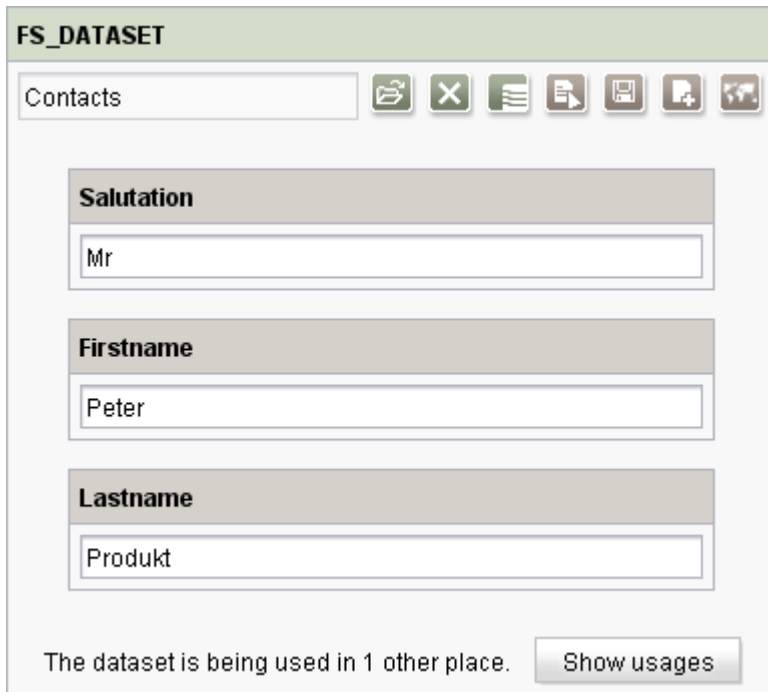
Add Link; click this icon to insert a new link. The procedure for setting a reference is analogous to the description of the "Link Input" input component. (See Chapter 10.23 page 291).

Delete Link; click this icon to delete the selected link from the link list.

Move Link Up; click this icon to move the selected link up in the list by one position.

Move Link Down; click this icon to move the selected link down in the list by one position.

10.25 Data set selection (FS_DATASET) (from V4.2)

Figure 10-65: Input component – Data set selection

> *This input component has in FirstSpirit Version 4.2 the status "Under development" and will be released officially in a later FirstSpirit version . For further information please refer "FirstSpirit Release Notes Version 4.2".*

Analogous to the CMS_INPUT_OBJECTCHOOSER input component, the new input component FS_DATASET is used to reference a dataset from the Content Store.

Unlike the selection option within the input component CMS_INPUT_OBJECTCHOOSER, FS_DATASET is not limited 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.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

This input component offers all editing options to the editor as the input component CMS_INPUT_OBJECTCHOOSER does (see Chapter 10.7 page 265). However, it is not possible to delete the data record from the data source using the icon if used in the Page Store. If it is used in the Content Store, this icon can be used for deletion (unlike in CMS_INPUT_OBJECTCHOOSER) in FS_DATASET as a default.

10.26 List new (FS_LIST) (from V4.2)

Figure 10-66: Input component – List new

> ❗ *This input component has in FirstSpirit Version 4.2 the status "Under development" and will be released officially in a later FirstSpirit version . For further information please refer "FirstSpirit Release Notes Version 4.2".*

This 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.

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 ("Section overview"). 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 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 ⬜ and ⬛ or ▶▼ at the left border of the section overview or by double-clicking the window frame.

If the project developer has defined a preset default value for the input component the 🔄 "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 page 386).

🔲 Add Section; click this icon to add a new section to the list. To do this, a section template for the new section can be selected first from a drop-down list:
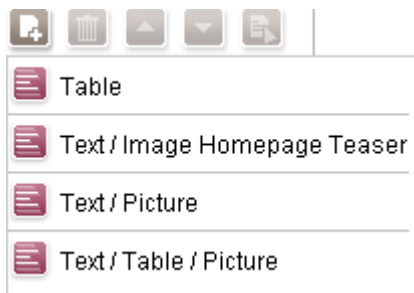
Figure 10-67: Select section template

The section in which the content can be edited is shown underneath the list display.

🗑 Delete Section: click this icon to delete the selected section from the section list.

🔼 🔽 Move Section Up / Down: click this icon to move the selected section up / down in the list by one position.

📄 Edit section: click to show the selected section in the bottom area for editing.

In FirstSpirit Version 4.2R4 and higher this input component also includes the functions of the following input components:

- CMS_INPUT_CONTENTLIST
- CMS_INPUT_LINKLIST
- CMS_INPUT_SECTIONLIST
- CMS_INPUT_TABLIST

FS_LIST can also be used in the familiar way for these input components. In some cases other icons are used and the selection dialog familiar from other input components including search function.

Figure 10-68: Selection of data records with new list

The following icons are available, depending on the defaults set by the template developer and the application purpose of the input component:

Add (only for data records), click this icon to open a selection dialog to select a data record

New, in the case of data records a window opens when clicking this icon to enter a new data record, in the case of sections and links a selection dialog will open for selecting a correspondent template

Copy, in the case of data records a copy of the selected data record will be created in the related data source if using this icon and added to this list.

Remove, click this icon to delete the selected list element from the list

Delete (only for data records), click this icon to delete the selected data record from the list and from the related data source. This data record can not be re-selected later on.

Move up / down, click this icon to move the selected list element one position up or down

Edit, click this icon to open a window for editing the selected list element. In the

View mode, the icon is displayed.

View, if overview and detailed information are shown in the input component, use this icon to open or close the detailed view of the entry which is selected in the overview. If only the overview is shown, use this icon to display the details of the selected entry in a separate window.

Jump to, click this icon to switch to the Content Store, in order to edit the selected data record there.

Open in separate window, use this icon to open the selected list entry in a separate window

In addition, it is also possible to edit the lists using a context menu or keyboard shortcut. The same functions are available here as provided by the icons:
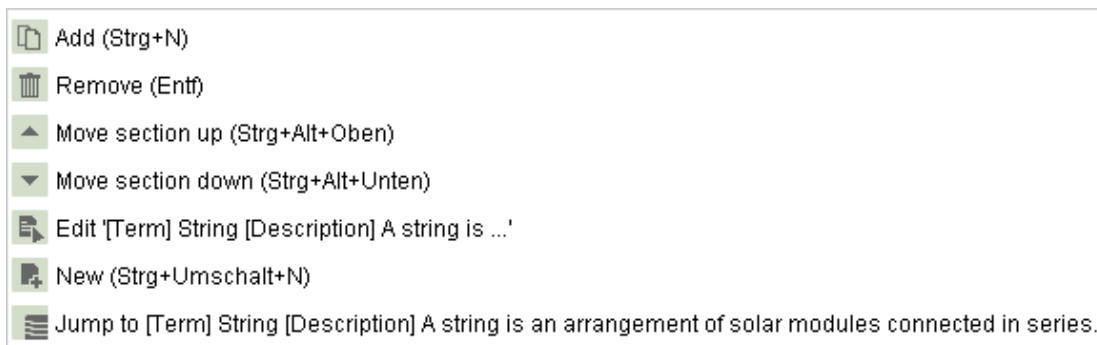
Figure 10-69: FS_LIST – Context menu

10.27 Reference selection (FS_REFERENCE) (from V4.2)

Figure 10-70: Input component – Reference selection

> **!** *This input component has in FirstSpirit Version 4.2 the status "Under development" and will be released officially in a later FirstSpirit version . For further information please refer "FirstSpirit Release Notes Version 4.2".*

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. Depending on the project developer's predefined settings, any object type can be selected. References to media (pictures and files), to page references (Site Store) and each other store and, e.g. to other folders.

If the project developer has defined a preset default value for the input component the "Accept Default Value" icon is located in the top right-hand corner. (For further information, see Chapter 11.10 from page 386.)

Reference: The input component displays a selected reference with object symbol, release status and the time and originator of the last change.

 Click this icon to open a selection dialog:
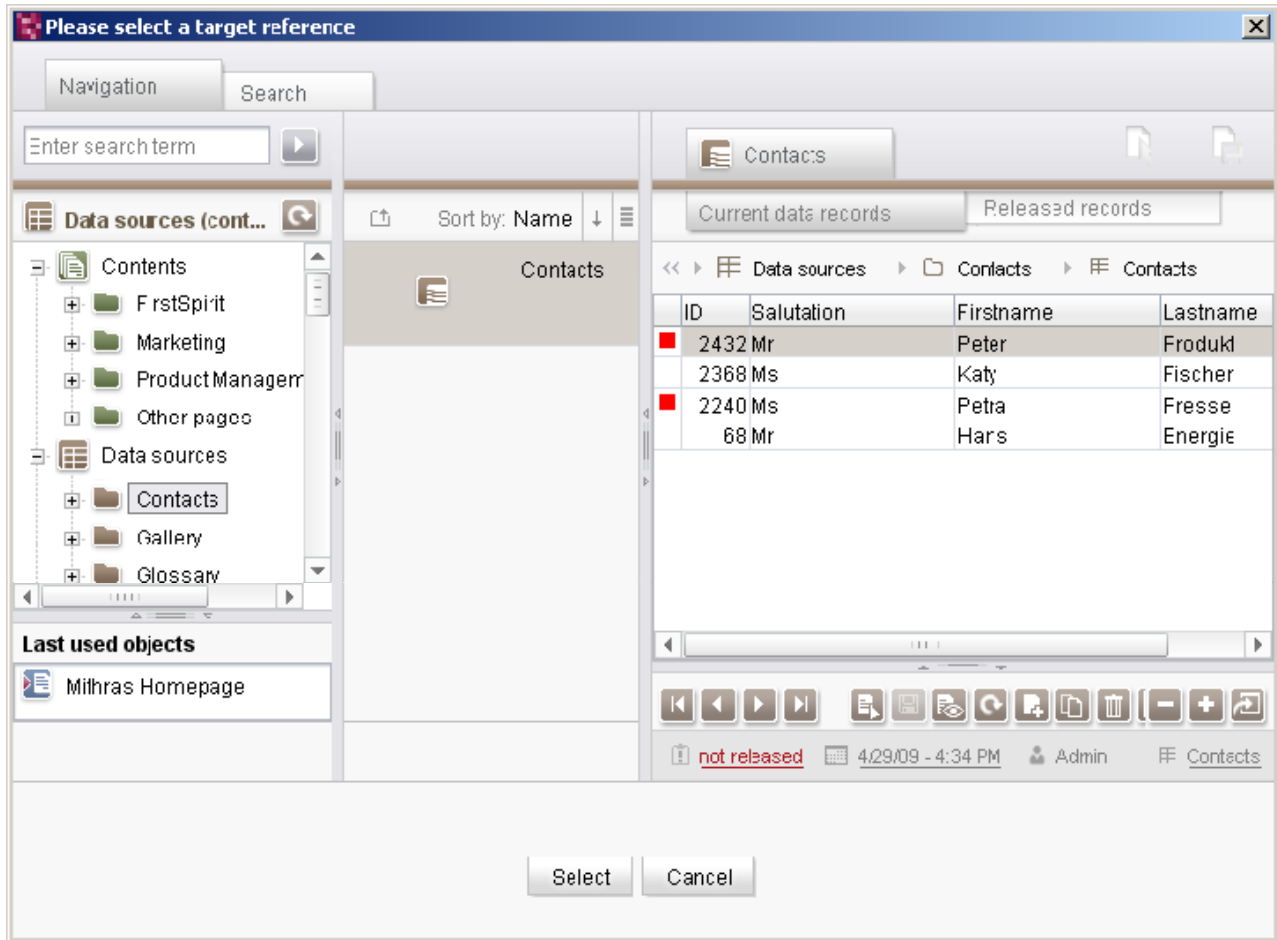
Figure 10-71: Select Reference (here: Content Store)

Depending on the project developer's predefined settings, between one and all stores and folders are displayed here, which can be selected as a reference. Remote projects can also be displayed. For details of navigation and handling the dialog, cf. also 11.8 page 360.

Click this icon and, depending on the configuration of the input component, media can be uploaded into the Media Store of the local or of a remote project. For details of uploading, see Chapter 10.3.2 page 255.

Click the icon to jump to the selected medium in the respective store.

Click the icon to remove the selected reference from the input component.

Section: Depending on the selected reference type a drop down list will be shown from which a section of the target page can be selected.

10.28 Button (FS_BUTTON) (from V4.2R4)

Depending on the template developer's settings, the new input component FS_BUTTON represents a button

FS_BUTTON

Figure 10-72: FS_BUTTON – button

an icon
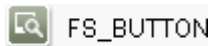
FS_BUTTON

Figure 10-73: FS_BUTTON – icon

or a simple link

FS_BUTTON

Figure 10-74: FS_BUTTON – link

When clicked a function is performed, which can be defined by the template developer. Depending on the configuration of the input component, the function can also be performed when an object (from FirstSpirit, files from the local workstation, etc.) is dragged onto this control using the mouse.

11 General Use

Several important features of the client are explained in greater detail in the following.

11.1 Task list

The task list manages all tasks issued in FirstSpirit. Tasks are workflows which each user can set for themselves or for other users. The task list can be opened in many ways ("Tasks - task list" menu, "Show task list" toolbar or CTRL + T).

The **Open tasks** tab contains all the tasks to be dealt with by the currently logged on user. It does not matter whether the tasks were assigned by the user themselves or by another user.

The **Initiated tasks** tab contains all tasks assigned by the currently logged in user. It does not matter whether the tasks are to be carried out by the user themselves or by another user.
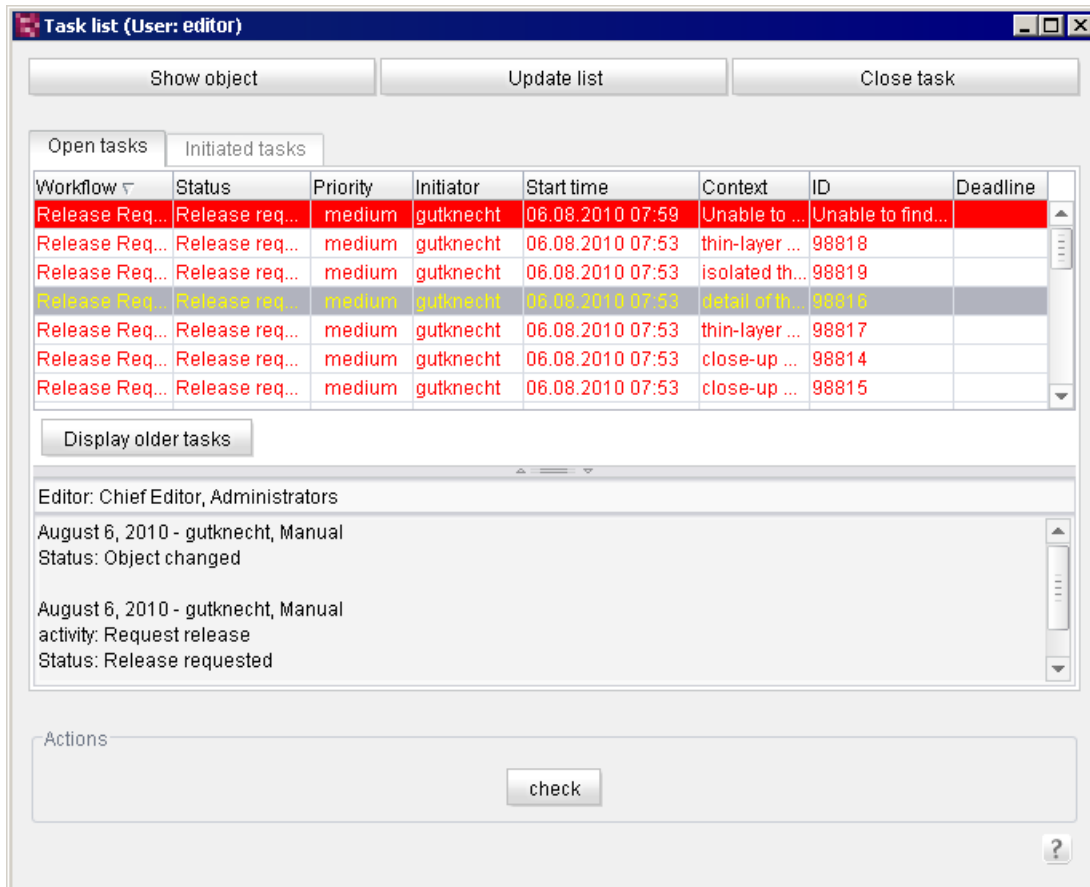
Figure 11-1: Task list in Version 4.2 and higher

In **FirstSpirit V4.2R4** and higher, for performance reasons, only 25 tasks are initially displayed in the task list on the "Open tasks" and "Initiated tasks" tabs. If more tasks exist, they can be shown by pressing the **Display older tasks** button.

The tasks are marked in colour for improved clarity. Black text means that the user is not directly selected as the modifier for this task. Red lettering means that the user is either directly selected as the modifier or belongs to a group that is the selected modifier.

In **FirstSpirit V4.2R4** and higher, invalid tasks, which result e.g. because an object on which a workflow is active is deleted, are visualised in the task list with a red background. These cannot be forwarded, but can only be closed using the "Close task" button. If the task can be repaired, e.g. if the deleted object for which the workflow still exists is restored, the "Repair task" button is shown in the actions area. Perform this action to reset the task, the status colour and the write protection.

The **Display object** button is used to set the focus in the tree structure of FirstSpirit JavaClient on the corresponding element of the selected task.

If further tasks have been added while the task list was open these new (or changed) tasks are not automatically updated. Click the **Update list** button to synchronise the task list of the FirstSpirit Server with the open task list.

Use the **Close task** button to close tasks without running through the corresponding workflow. This might be necessary, for example, if a task has become invalid (e.g. by deleting the object on which the workflow is active).

In FirstSpirit-Version 4.1 and higher open tasks are also displayed on the project entry page (see chapter 1.3.1 page 24).

11.1.1 Open Tasks tab

All tasks to be dealt with by the currently logged in user are displayed in this area. It does not matter whether the tasks were assigned by the users themselves or by another user.

11.1.2 Initiated Tasks tab

All tasks assigned by the currently logged in user are displayed in this area. It does not matter whether they sent the tasks to other users or to themselves as a reminder.

11.2 The Message Board

There are message boards in several places in the client. The message board function is identical everywhere. The message boards are located in the following places in the individual Stores:

- Page Store:
  Store level, Folder level
- Site Store:
  Store level and Folder level, each in the Messages tab
- Media Store:
  Store level and Folder level, each in the Messages tab
- Template Store:
  Store level, in the Page templates areas, Section templates, Mappings, Format templates and Folder level
- Content Store:
  Store level, Folder level

The purpose of these communication platforms is to exchange context-sensitive information between individual users. E.g. here at the level of a folder in the Page Store, editing notes can be forwarded which are to apply to all pages in this folder. Each editor has the opportunity to reply to these and to submit supplementary comments. All messages are stored as can be seen in the usual tree structure so that it is easy to follow a discussion from its start through to the final reply.

If the message board is still empty at the beginning or if you want to create a completely new message, click the word "Message Board" to activate the "New Message" button. If messages have already been placed on the message board, click within the tree structure to open the message content. To this end the subject appears in bold writing in the bottom part followed by details of the author and the date the original message was created. The actual message content is displayed in the white window below this. If you wish to reply, click the "Reply" button. A window then opens in which you can change the subject, which must be extended by the prefix "AW:" and below that you can enter your messages.

Click the "Delete" button to delete the currently highlighted message as well as all its linked answers without any further confirmation prompt.

11.3   Metadata

Metadata is additional information available for an object in FirstSpirit. Such information can be, e.g. "Last Change Date", "Last Editing Editor", "Released by", "Version Number", etc.

Some metadata assigned by the system can be opened at each node in the tree structure with the hotkey "**ALT + P**". E.g. the number of integrated media can be ascertained in this way.

Apart from these metadata assigned by the system it is possible to define further metadata which is managed by the editor. The editor can make changes to this user-defined information in the Metadata tab.

Special permissions are required to edit metadata; these must be assigned in the Permission Management window (see Chapter 13.1.4.11 page 460). Changes can only be made to the metadata if they are blocked.

Project-specific data structures can be defined for each project with which a hierarchical description of the objects can be entered in the Page, Site, Media and content stores.

Evaluation of the project data can take place in three different ways:

1. When a metadata variable is read out the selected node is checked. If this variable is empty and empty string is output, otherwise the relevant variable value.
2. When a metadata variable is read out the selected node is always checked first. If this variable is empty the next higher node in which this variable is filled with a value is searched for in the hierarchy and this value is output.
3. When a metadata variable is read out each node in the hierarchy up to the selected node is checked, starting from the store root. The values of the hierarchy nodes are output in order and are separated by a separator.

In the case of **section references** (see Chapter 4.1.3 page 114), the respective source section or the object chain of the source section will be used to determine the meta information for 1. and 2., and the object chain of the section reference for 3.

The template developer decides which structure a project's metadata has and which evaluation methods are to be applied; this can be read up in the relevant project documentation.

Each node in the tree structure for which the editor has given metadata is denoted by a green dot, in the Look & Feel "LightGray" from FirstSpirit Version 4.1 by the icon i.

11.4   Drag & Drop FirstSpirit objects (from V4.2)

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

11.4.1   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.

11.4.2   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 11-2).
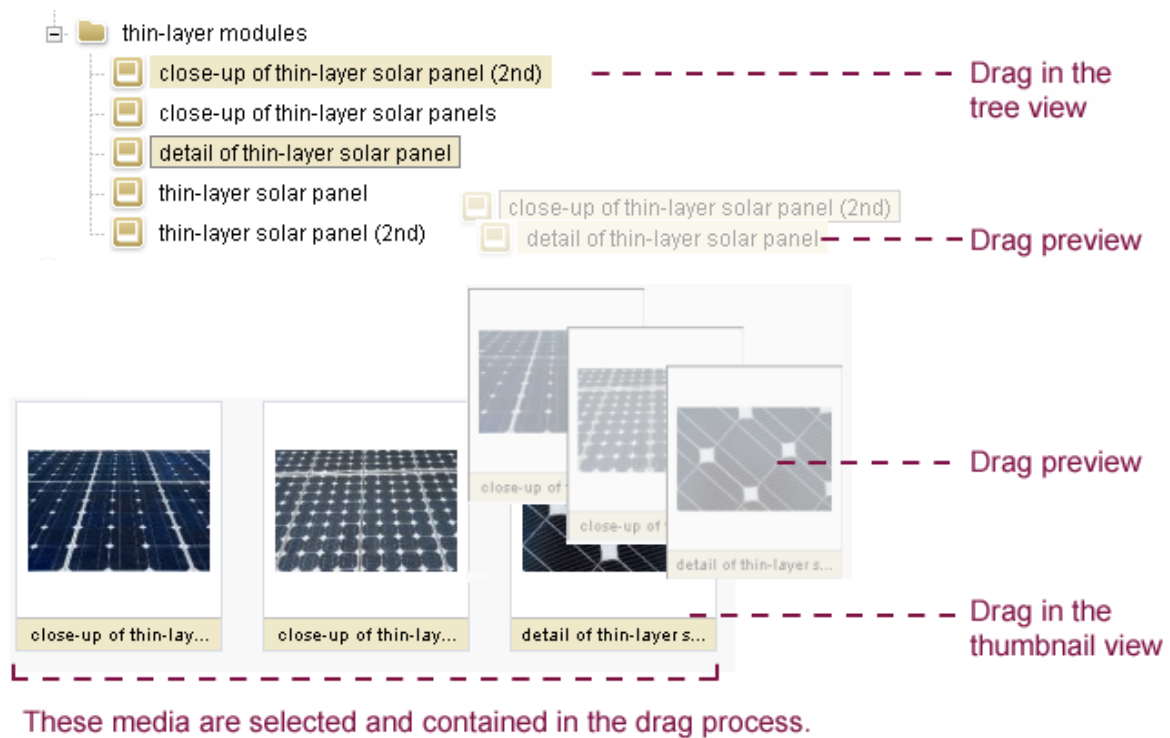
Figure 11-2: Drag action in the tree and in the thumbnail overview

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 11-3):
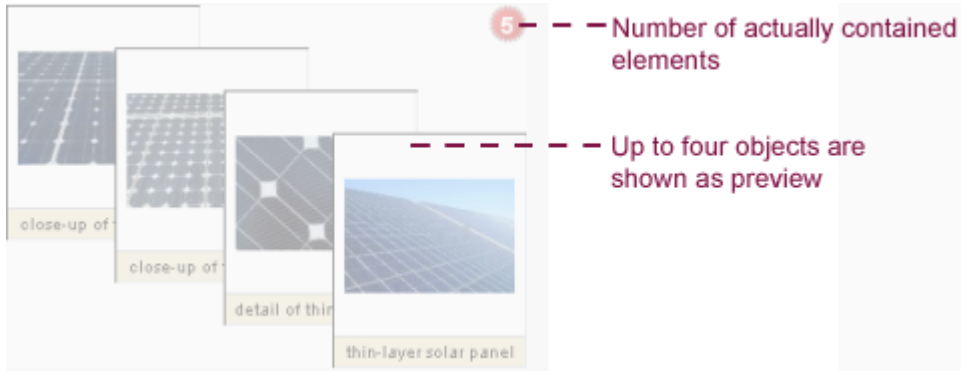
Figure 11-3: Drag action with more than four elements

11.4.3 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).
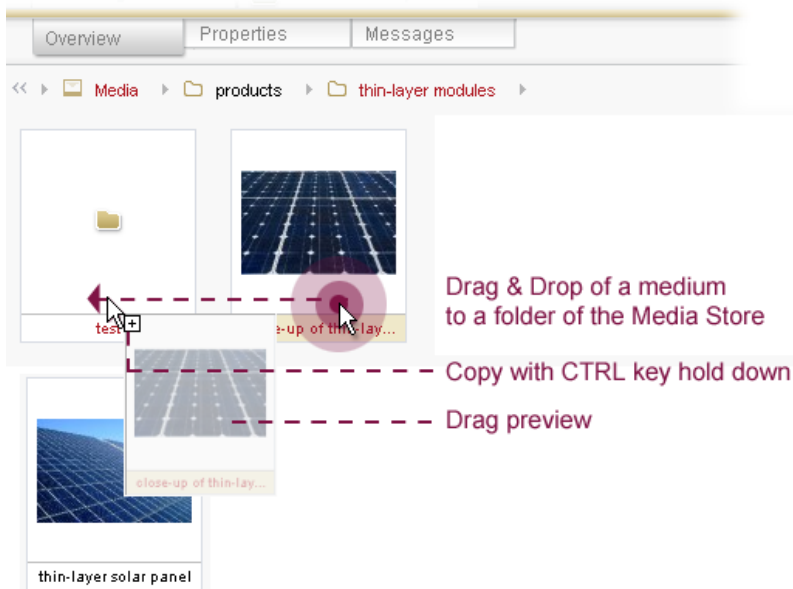
Figure 11-4: Drag & Drop media in the thumbnail view

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.

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.

11.4.4   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 1.3.2.1 page 25).

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 11-5).
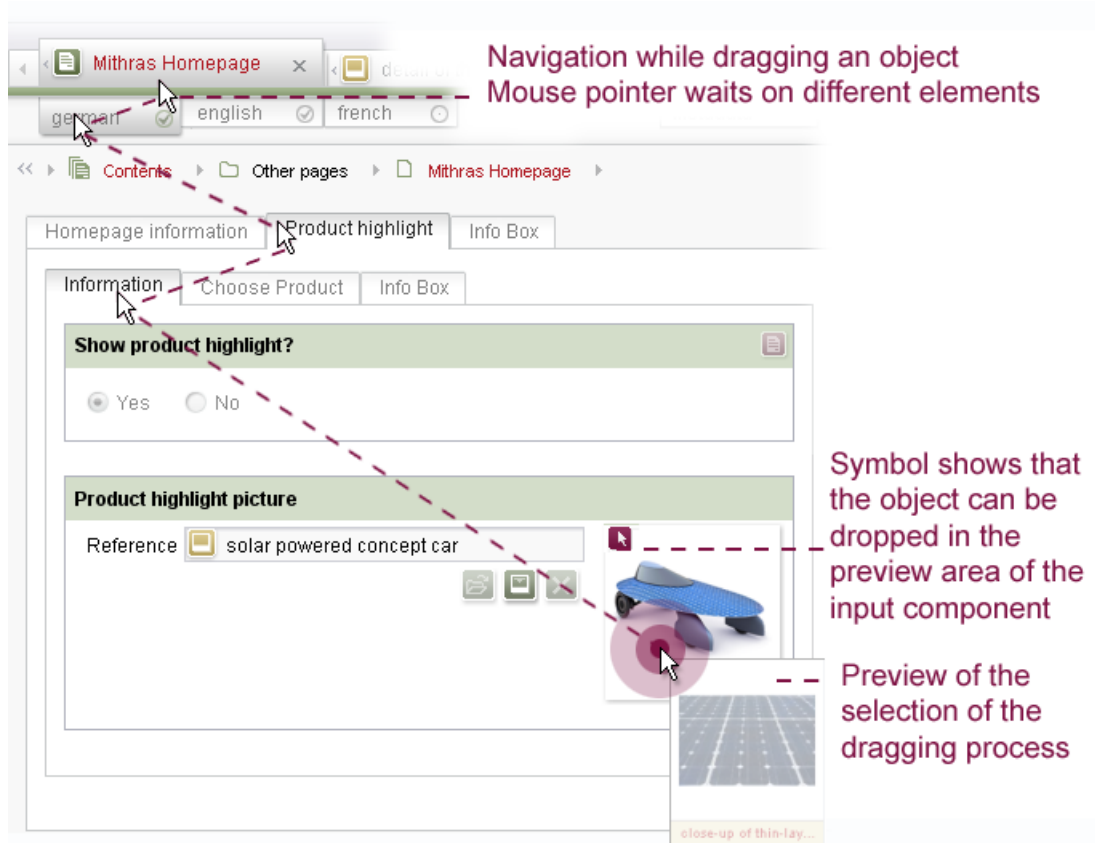
Figure 11-5: Navigation during a drag action

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.

11.4.5 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.

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"

(see Chapter 11.4.4 page 321).

11.4.6 Drag & Drop from the search dialog into JavaClient

The new (non-modal) search dialog for the quick text search (see Chapter 3.3 page 106) 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 3-60).

11.4.7 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):
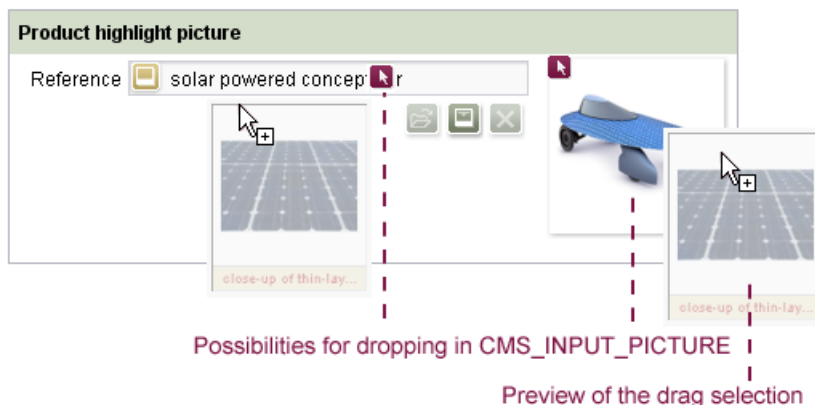
Figure 11-6: Drop into a picture input component

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.

For information about Drag & Drop functions in the input component CMS_INPUT_CONTENTLIST for the "Media Gallery" functionality see Chapter

11.4.8 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 (see Chapter 11.4.4 page 321).

11.4.9 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.

11.4.10 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 (see also Chapter 5.4 page 147). 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.

11.4.11 Drag & Drop from the Integrated preview

You can also drag media using Drag & Drop from a page of the Integrated Preview into the working space of the Media Store. In this case, the function "Content highlighting control" should be deactivated in the menu "View".

If you drop the picture onto one of the resolutions of an existing picture, the dropped picture will be directly inserted for the selected resolution. If you drop the picture onto a folder of the Media Store, you will first be asked for the display and the reference name of the new picture, then the new picture will be inserted into the selected folder of the Media Store.

11.5   General information on input components

Figure 11-7: Examples of input components

Input components are used to file and manage editorial content in FirstSpirit. Each input component has a title (framed in red in Figure 11-7). It is colour coded, depending on the status:

- **Headline** : As a default, titles of multi-line components are shown in the colour of the respective store (e.g. Page Store: green). As a default, titles of single line components do not have any markings whatsoever.

- **Paragraph** : A title with a red background indicates that the component is not correctly filled or contains and invalid reference (see also Figure 3-19, page 68). Single line components with this status are framed in red.

- **Type of Link** : The pink coloured marking shows that the content of the input component is a preset default value (see Chapter 11.10, page 386). Single line components with default values are highlighted by a pink coloured frame and a pink coloured vertical bar on the left next to the title.

11.6 Working with the DOM Editor input component

The DOM Editor is provided for large formatted text inputs. Formatting can be defined both for highlighted text fragments and for whole sections. All functions of the DOM Editor icon bar, the context menus and the corresponding keyboard shortcuts (hotkeys) are explained in this manual.
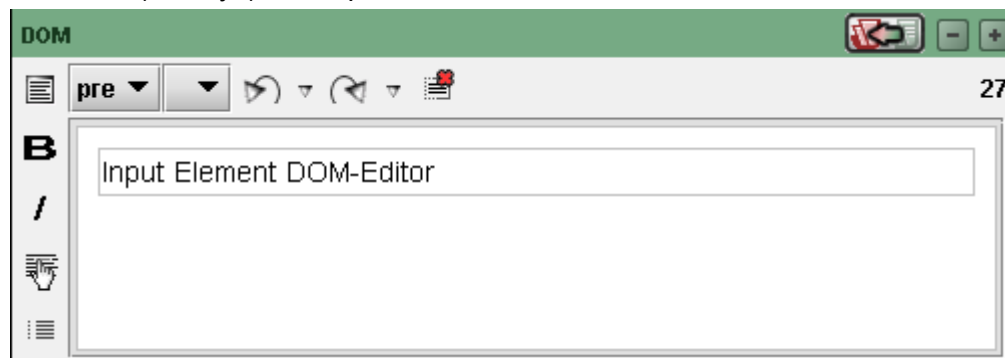
Figure 11-8: DOM_Editor input component

11.6.1 DOM Editor icon bar

The standard functions which can be opened via the icon bar have already been described in the DOM Editor Chapter under Standard Input Components. See Chapter 10.11 page 276.

11.6.2 DOM Editor context menus

11.6.2.1 Context menu for highlighted text

Paste
Copy
Cut

Figure 11-9: DOM Editor – Context menu on highlighted text

Paste: Open this function to paste text fragments from the clipboard in the cursor's current position in the section.

Copy: Open this function to copy the selected text of the section into the clipboard.

Cut: This function is used to cut the selected text of the section and copy it into the clipboard.

11.6.2.2 Context menu within a list

Change list type
Move item one level up
Paste
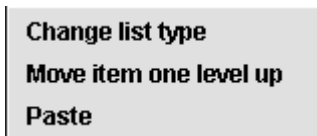
Figure 11-10: DOM Editor – Context menu within a list

Change List Type: This function can be used to change the list type.

Change list type
Type   Numbered ▼
User-defined text [   ]
                OK   Cancel

Figure 11-11: Change List Type

The combo box can be used to select the required type. Depending on the list type it is also possible to specify which number or letter the list should start with or which picture of the selected list type is to be displayed on the website.

Move item one level up: This function can be used to move (classify) the

highlighted list item up by one level.

Paste: Use this function to paste text fragments or items in a list from the clipboard into the cursor's current position in the list. Line breaks are inserted here as new list items.

11.6.2.3 Context menu on formatting

Remove format
Paste
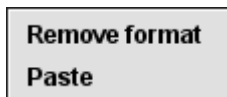
Figure 11-12: DOM Editor – Context menu on formatting

Remove format: Open this function to remove the formatting for the selected text area. The text is retained.

Paste: Use this function to paste text from the clipboard at the position of the cursor.

11.6.2.4 Context menu on references

Edit link
Remove link
Paste
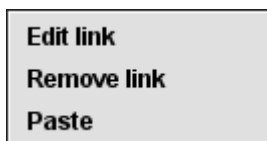
Figure 11-13: DOM Editor – Context menu on references

Edit link: Open this function to open the input window of the link type used so that the settings made can be edited. For a detailed description of the settings for the various link types, see Chapter 10.23 page 291.

Remove link: Open this function to remove the selected link. The link text is retained.

Paste: Use this function to paste text from the clipboard at the position of the cursor.

11.6.3 Spell check in the DOM Editor

The spell check for a project must be configured by the administrator. This requires the installation of a module (SpellService) on the server. The spell check is configured to be project-specific, the examples shown here can therefore differ from the display in the project.
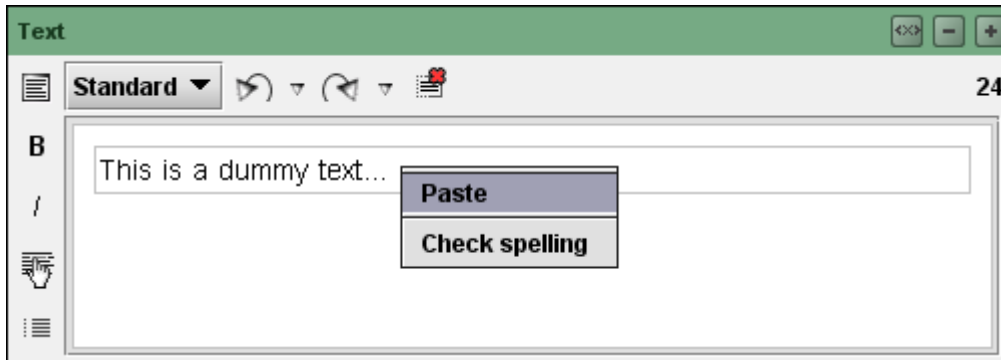
Figure 11-14: Check spelling

If spellchecking is configured for a project it can be opened in the DOM and DOM Table input components using the context menu "Check Spelling" entry.

All unknown words are then highlighted in red in the input component. All words which do not exist in the language-dependent dictionary lists configured for the respective project by the administrator are "unknown". Either the spelling of the word is unknown, e.g. due to a typing error or the spelling is correct and the word is simply not included in the list used.
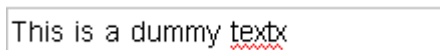
Figure 11-15: Display unknown words in the input component

The user can open the context menu by right-clicking the word highlighted in red:
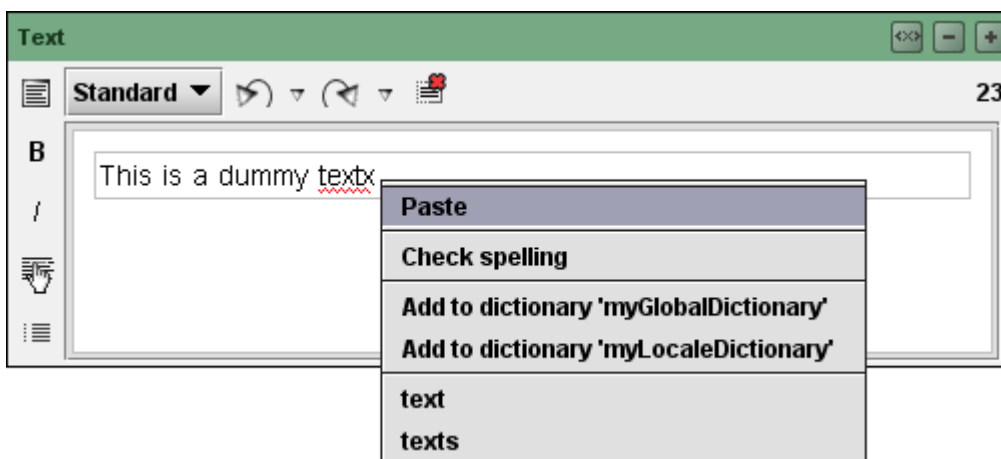
Figure 11-16: Check spelling context menu

Add to dictionary: Apart from the suggested improvements, depending on the user's permissions and depending on the configuration of the dictionary (this must be

defined as "changeable"), words highlighted in red can be added to one or several dictionaries. In future these words are no longer highlighted red.

A differentiation is made between so-called "Global Dictionaries" and "Local Project Dictionaries":

▪ Changes to global dictionaries affect all projects on the server (if the global dictionaries are used in the project configuration).

▪ Changes to local project dictionaries always affect the current projects only. This means a word added in Project A is again highlighted as an unknown word in Project B.

Depending on the project configuration and the permissions of the user, several different dictionaries (or none) can be available.

Ignore all: The spell check will be ignored for all occurences of the word in the input component. The word will no more be highlighted in red, but it will not be added to the dictionary. If the spell check will be started anew the word will be highlighted again.

Suggested improvements: If similar words are found in the word lists used, suggested improvements are displayed in the bottom part of the context menu. Click the required suggestion to replace the unknown word with the suggestion from the dictionary.

11.6.4 Lists: Generating lists

11.6.4.1 Generate a new empty list

Click the **Insert List** ☰ icon to generate a new, empty list.

Figure 11-17: Generate new, empty list

The cursor is then in the new empty listpoint (bullet point).

11.6.4.2 Add new bullet point
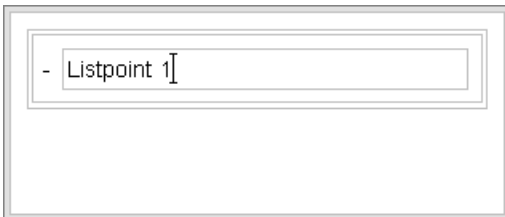
Press <RETURN> at the end of a bullet point to add a new empty bullet point.

Figure 11-18: Add bullet point 1

Figure 11-19: Add bullet point 2

The cursor is then in this empty bullet point.

11.6.4.3 Add several new bullet points
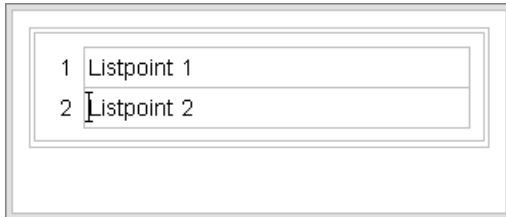
Press <RETURN> at the start of a bullet point which is not empty to add a new empty bullet point in front of this bullet point.

Figure 11-20: Several bullet points 1
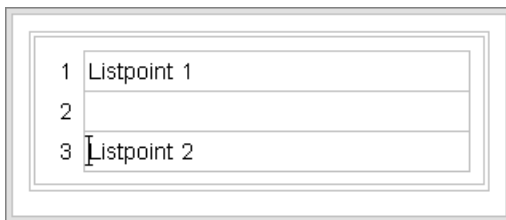
Any number of empty bullet points can be generated in this way.

Figure 11-21: Several bullet points 2

The cursor is then still at the start of the bullet point which is not empty.

11.6.4.4 Add new bullet point with content
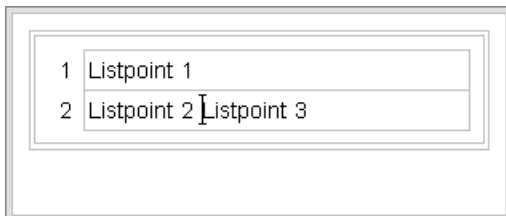
Press <RETURN> within a bullet point to generate a new bullet point from the rest of the line.

Figure 11-22: Add bullet point with content 1
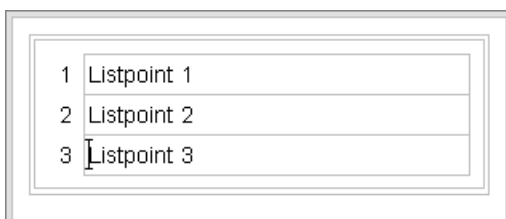
Figure 11-23: Add bullet point with content 2

The cursor is then at the start of this new bullet point in front of its content.

11.6.4.5 Add new bullet point to a nested list
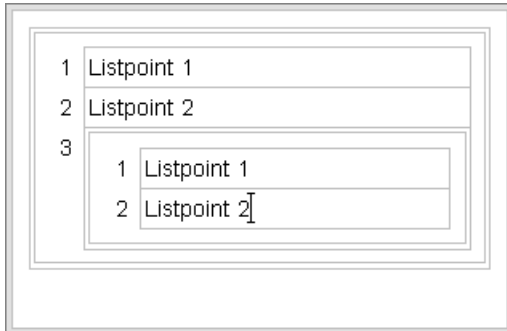
Press <RETURN> at the end of a nested (indented) bullet point to add a new empty bullet point with the same nesting depth.

Figure 11-24: Add nested bullet point 1
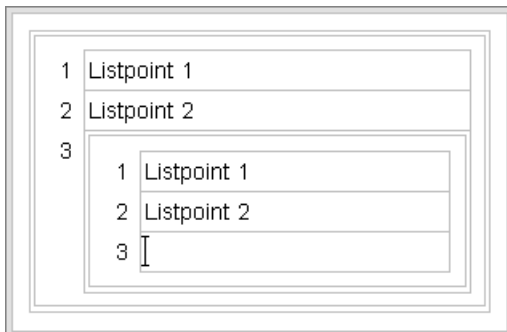
Figure 11-25: Add nested bullet point 2

The cursor is then in the new empty listpoint (bullet point).

11.6.4.6 Add new bullet point behind a nested list
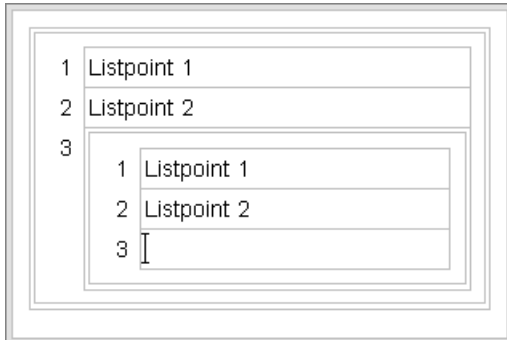
Press <RETURN> in an empty bullet point on the 2nd list level to move the empty nested bullet point to one level higher (i.e. indent is increased).

Figure 11-26: Increase bullet point level 1
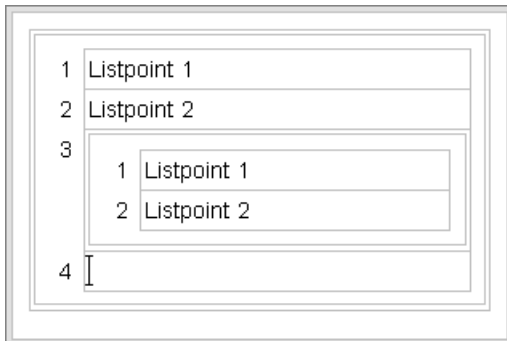
Figure 11-27: Increase bullet point level 2

The cursor is then still in the empty bullet point.

11.6.5 Convert selected text into list

11.6.5.1 Complete highlighting (selection) within a section

If the whole content of an individual section is highlighted it can be converted into a list by clicking the **Insert List** icon (≡).

Figure 11-28: Convert complete section into list 1
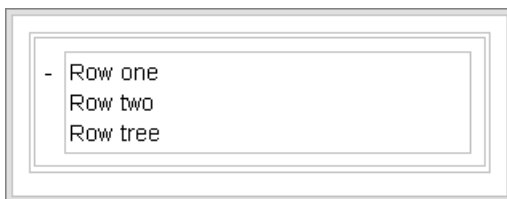
Figure 11-29: Convert complete section into list 2

The list consists of one bullet point.

11.6.5.2 Partial highlighting (selection) within a section
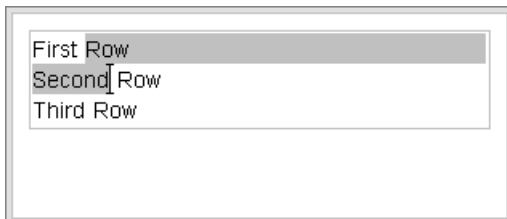
If the content of an individual section is partially highlighted this highlighted content can be converted into a list by clicking the **Insert List** icon (≡).

Figure 11-30: Partially convert section into list 1
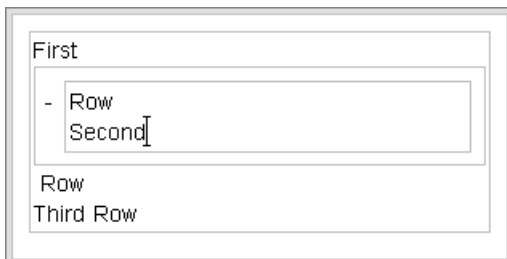
Figure 11-31: Partially convert section into list 2

The list consists of one bullet point. The text fragments which are not highlighted are

kept in the same section in front of or behind the list.

11.6.5.3 Complete highlighting of several sections
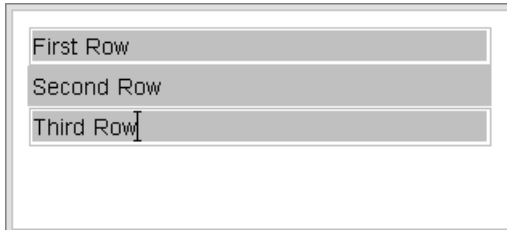
If the whole content of several sections is highlighted it can be converted into a list by clicking the **Insert List** icon (≡), whereby each section end is interpreted as a new bullet point.

Figure 11-32: Convert several sections into list 1
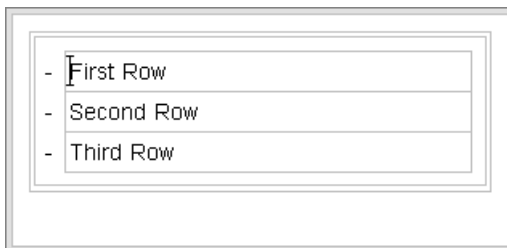
Figure 11-33: Convert several sections into list 2

If there are three existing sections the list then also consists of three bullet points.

11.6.5.4 Partial highlighting of several sections
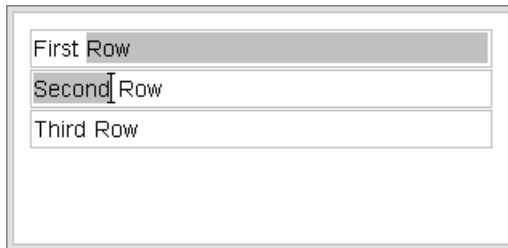
If the content of several sections is partially highlighted this highlighted content can be converted into a list by clicking the **Insert List** icon (≣).

Figure 11-34: Partially convert several sections into list 1
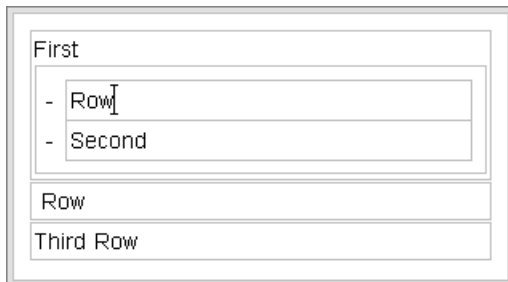
Figure 11-35: Partially convert several sections into list 2

If a section change is highlighted the list consists of two bullet points. The text fragments within a section which are not highlighted are retained in front of or behind the list. Text fragments without section change in the same section and text fragments with section change in a new section.

11.6.6   Exit / interrupt list

11.6.6.1   Exit list at the end

Press <RETURN> in an empty bullet point to remove the empty bullet point and replace it with a new, empty, standard section.

Figure 11-36: Exit list at the end 1
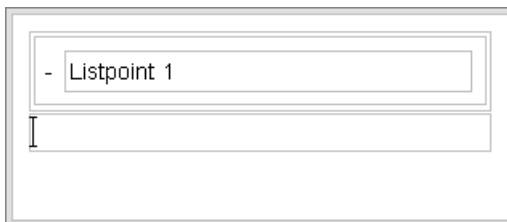
Figure 11-37: Exit list at the end 2

The cursor is then in this empty standard section.

11.6.6.2   Divide list between two bullet points

To divide a list an empty bullet point must be inserted at the required position.
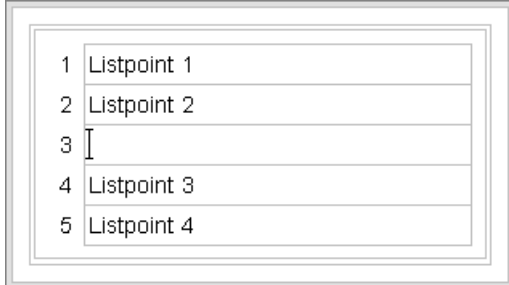
Then press <RETURN> to divide the list at this point and the numbering of the existing numbered points is reset accordingly.

Figure 11-38: Divide list 1
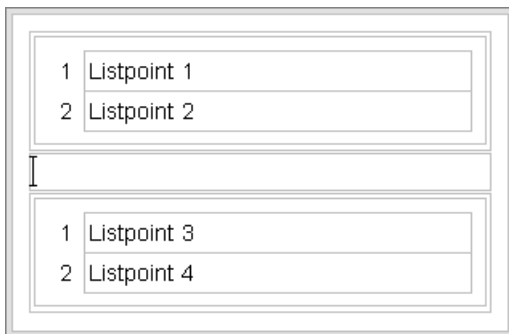
An empty standard section is inserted between the two lists.

Figure 11-39: Divide list 1

The cursor is then in this empty standard section.

11.6.6.3   Divide list at the end of a nested list
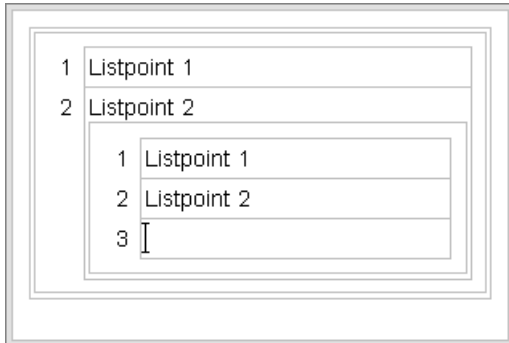
Press <RETURN> in an empty bullet point at the end of the 2nd list level to move the empty nested bullet point to one level higher (i.e. indent is increased).

Figure 11-40: Divide nested list 1
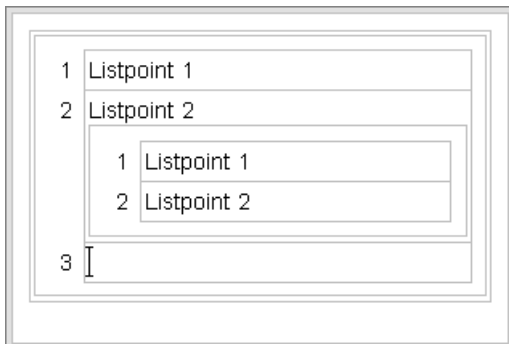
The numbering of the 1st list level is continued with the new numbered point.

Figure 11-41: Divide nested list 1

The cursor is then in the empty list point of the 1st numbered list level.

11.6.6.4   Divide list within a nested list
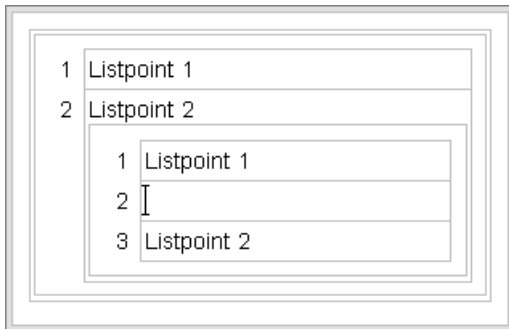
Press <RETURN> in an empty numbered point on the 2nd list level to divide the list at this point and move the empty nested numbered point to one level higher (i.e. indent is increased).

Figure 11-42: Divide list within a nested list 1
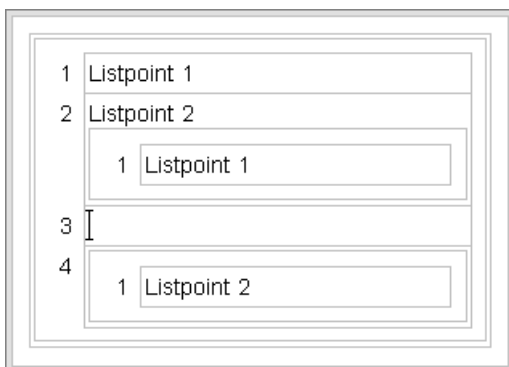
The numbering of the existing numbered points of the 2nd list level is reset accordingly. The numbering of the 1st list level is continued with the new numbered point and the second part of the nested list

Figure 11-43: Divide list within a nested list 2

The cursor is then in the empty list point of the 1st numbered list level.

11.6.7 Delete/merge lists

11.6.7.1 Merge a preceding section with the first list point
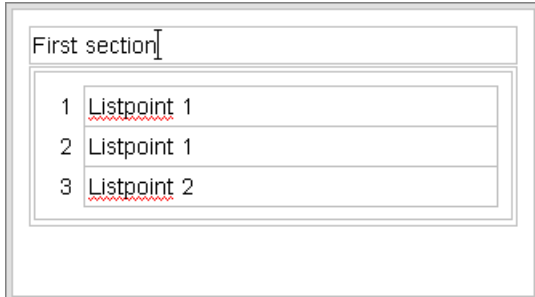
Press <DEL> at the end of the preceding section to merge the preceding section and the list to form one section.

Figure 11-44: Merge list with preceding section 1
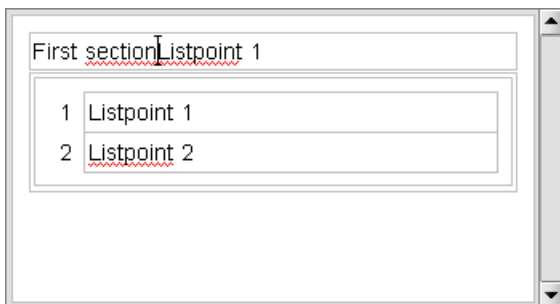
Figure 11-45: Merge list with preceding section 2

The cursor is then between the original section and list text.

11.6.7.2   Merging the last list point with the following section
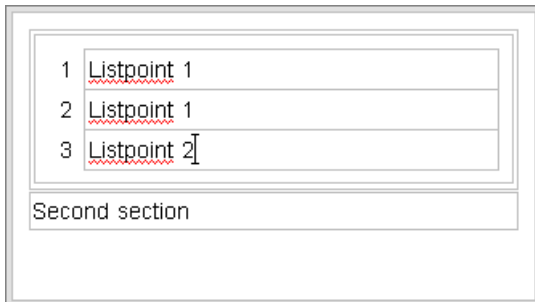
Press <DEL> at the end of the last list point to merge the list and the following section to form one section. The section text is written behind the text of the last list point.

Figure 11-46: Merge list with follow section 1
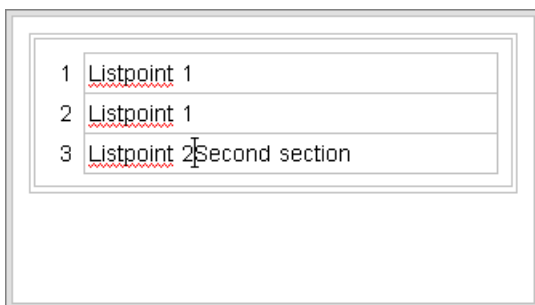
Figure 11-47: Merge list with follow section 2

The cursor is between the original list and section text.

11.6.7.3 Merging two list points
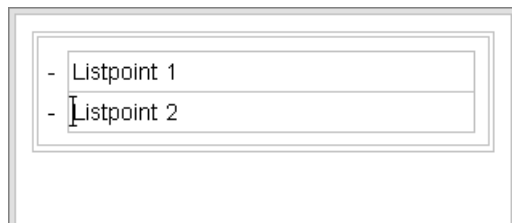
Press <BACKSPACE> at the beginning of the following bullet point to merge it with the preceding bullet point.

Figure 11-48: Merge two list points 1
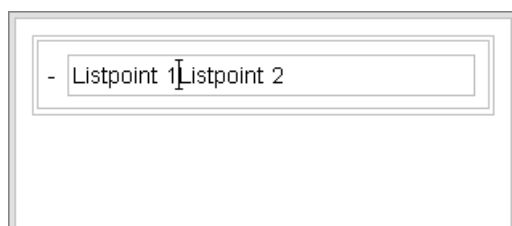
Figure 11-49: Merge two list points 2

The cursor is located between the original list text of the preceding and of the following bullet point.

11.6.7.4 Merging two lists
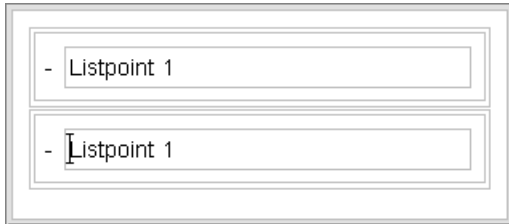
Figure 11-50: Merge two lists 1

If the two lists are each in a separate section, press <BACKSPACE> at the beginning of the second list to display both lists in one section.
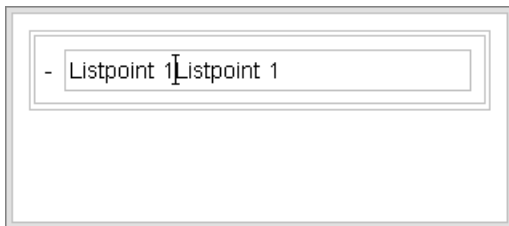
Figure 11-51: Merge two lists 1

Then press <DEL> to merge the two lists to form one list.
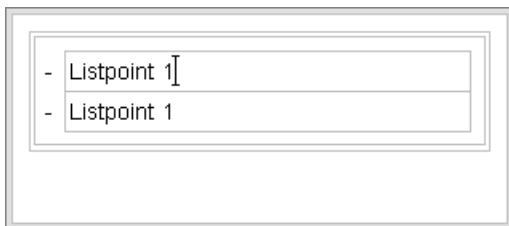
Figure 11-52: Merge two lists 1

The cursor is then at the start of the first bullet point of the original second list.

11.6.7.5 Merge the last numbered point of an inner list with the following numbered point

Figure 11-53: Merge inner list with the following numbered point 1
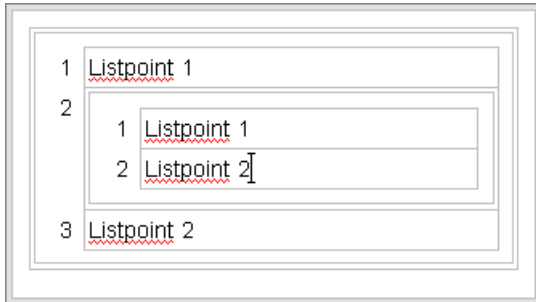
Press <DEL> at the end of the last numbered point of the 2nd level to write the list text of the 1st level behind the text of the preceding numbered point of the 2nd level.
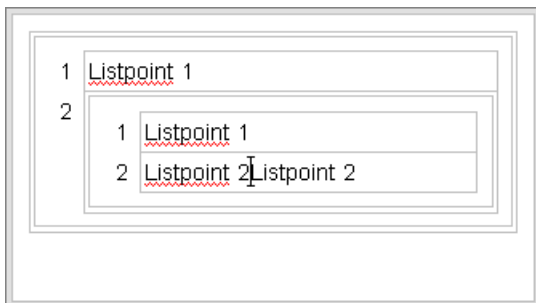
Figure 11-54: Merge inner list with the following numbered point 2

The cursor is located between the original list text of the 1st and 2nd level.

11.6.7.6 Deleting an empty bullet point
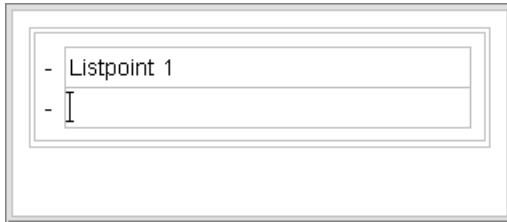
If the cursor is in an empty bullet point, this bullet point can be removed by pressing <BACKSPACE> or <DEL>.

Figure 11-55: Delete empty bullet point 1
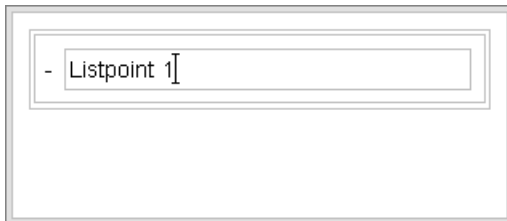
Figure 11-56: Delete empty bullet point 2

The cursor is then at the end of the preceding bullet point.

11.6.8 Copy/Move lists

11.6.8.1 Copy complete list
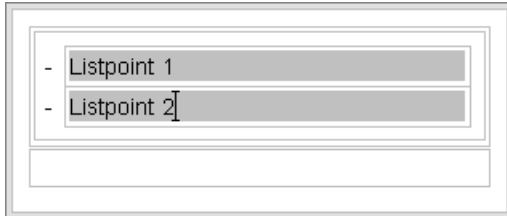
Figure 11-57: Copy list 1

To copy the whole list, highlight it and press <CTRL> + <C>. Then position the cursor in an empty section and press <CTRL> + <V>.
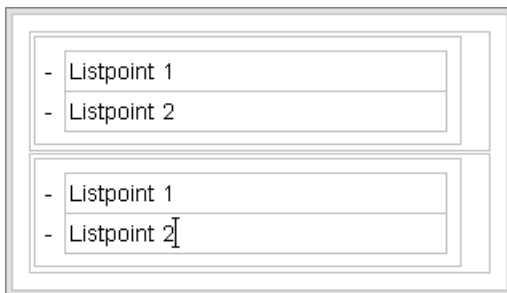
Figure 11-58: Copy list 2

The complete list is pasted in the empty section as a new list

The cursor is then at the end of the last bullet point.

11.6.8.2   Copy partially highlighted list

To copy part of the list, highlight it and press <CTRL> + <C>. Then position the cursor in an empty section and press <CTRL> + <V>.
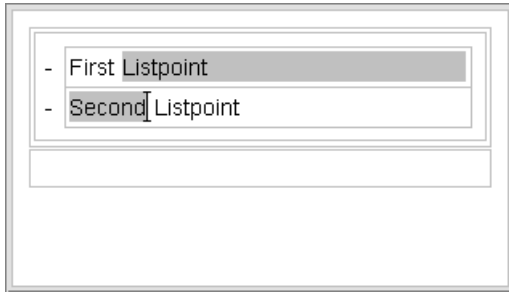
Figure 11-59: Partially copy list 1
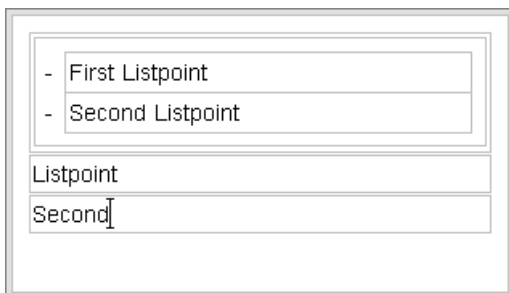
The selected text is pasted in the empty section.

Figure 11-60: Partially copy list 2

The cursor is then at the end of the last bullet point. If the highlighting extends over several bullet points then a new section is created for each bullet point.

11.6.9 Insert links

If the ▩ icon is used to insert a link at the cursor's current position a window opens for selecting the link type.
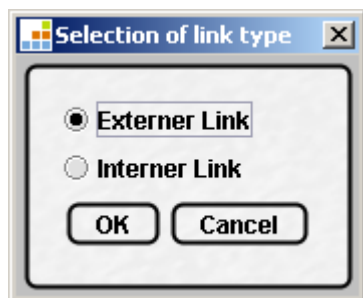
Figure 11-61: Select link type

The project developers specify which types of links are available. After selecting the link type an input window is displayed whose content depends on the link type.

The procedure for setting a reference is analogous to the description of the "Link Input" input component. (See Chapter 10.23 page 291).

11.6.10 Tables in the DOM Editor (from V4.1)

> ❗ *This function is released for FirstSpirit Version 4.1 and higher only. Screenshots are therefore displayed in the new "LightGray" Look & Feel. The display in the "Classic" Look & Feel can differ slightly.*

From Version 4.1 tables can be integrated in the text flow (so-called "inline tables"). Any number of layout design options can be made available to the editor, down to cell level.

> ❗ *Whether the Inline function is available in the DOM Editor or not depends on the template developer's default settings. If the* 🔲 🔲 🔲 🔲 🔲 🔲 🔲 🔲 🔲 *icon bar is available in the DOM Editor, the Inline Tables function is also available.*

To insert a table into the DOM Editor, the cursor is placed at the position within the DOM Editor at which the table is to be inserted. Click the 🔲 icon to insert a new table. All the available table format templates are displayed in the window that opens.
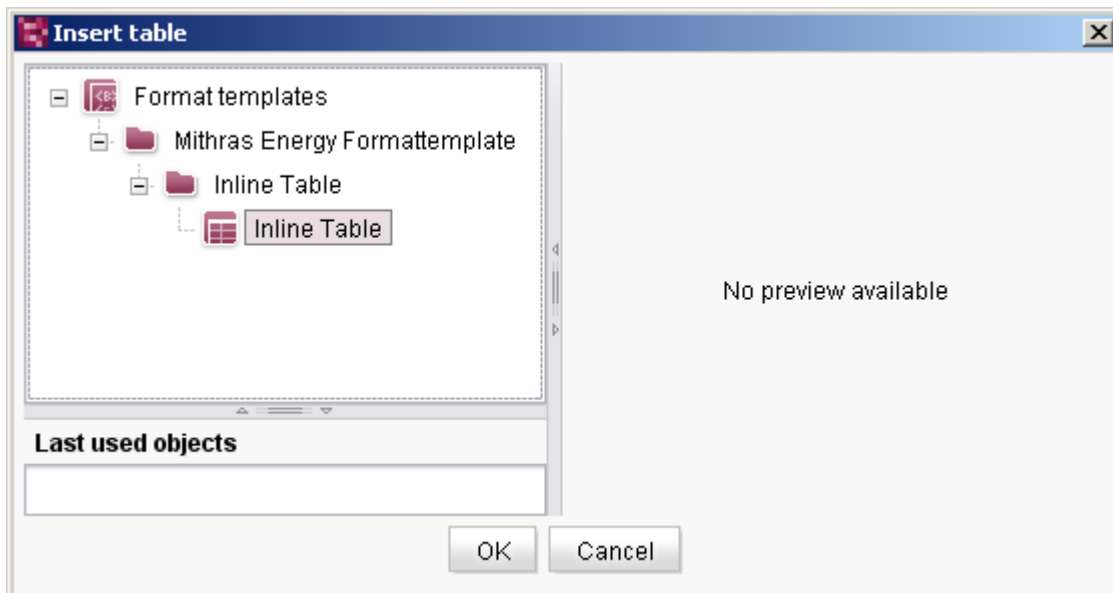
Figure 11-62: Insert table – Select table format template

The required table format template can be selected from the tree structure and the

selection confirmed with [OK].
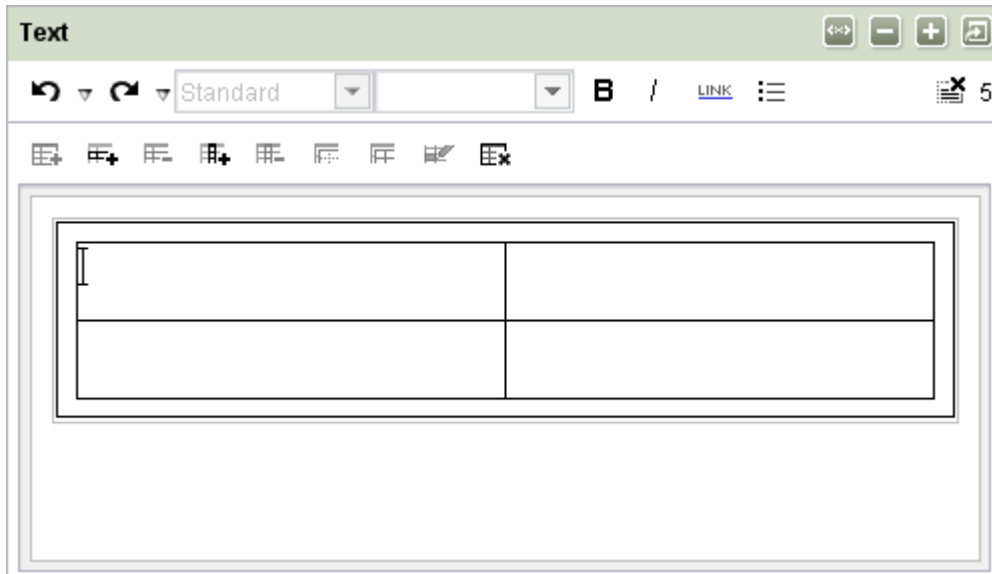
Figure 11-63: Table inserted

The number of rows and columns and the formatting of the inserted table depends on the template developer's settings.

11.6.10.1 Editing inline tables

The functions available for editing the inserted table depend on the preset values of the template developer, e.g. whether and how many rows and columns can be added or deleted, what format they have and which cells may be additionally formatted. The following icons above the table are only active in the cells which may be edited according to the developer's specifications. To do this, the cursor must be placed in the required cell:

Add Row: click this icon to add a row to the table below the cell in which the text cursor is positioned. As many rows as necessary can be added, until the maximum number preset by the developer is reached. The icon is then disabled, and no further rows can be added.

Delete Row: click this icon to delete from the table the row including contents, in which the text cursor is positioned. As many rows as necessary can be deleted, until the minimum number preset by the developer is reached. The icon is then disabled, and no further rows can be deleted.

Add Column: click this icon to add a column to the table to the right of the cell in

which the text cursor is positioned. As many columns as necessary can be added, until the maximum number preset by the developer is reached. The icon is then disabled, and no further columns can be added.

⊞ Delete Column: click this icon to delete from the table the whole column including contents, in which the text cursor is positioned. As many columns as necessary can be deleted, until the minimum number preset by the developer is reached. The icon is then disabled, and no further columns can be deleted.

⊞ Merge Cells: Several rows positioned next to each other can be marked with the mouse cursor and joined together by clicking this icon. The content of the marked cells is retained and is strung together from the cells, from left to right and from the top down. The formatting of the top left-hand cell is applied to the merged cell.

⊞ Split Cells: click this icon to separate previously merged cells again. The content of the merged cells is retained and is moved into the top left-hand cell. The format of the individual cells is reset to what it was before the merge.

⊞ Cell Properties: click this icon to open a window with properties which can be assigned to the current cell. Several cells to which the same property can be applied according to the template developer's specifications can be simultaneously selected and edited.
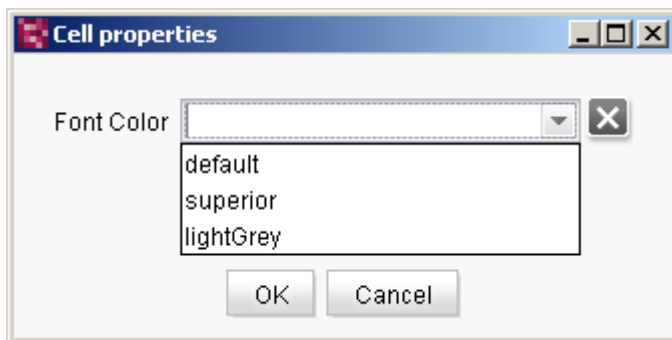
Figure 11-64: Example: Table format template – Cell properties

The required property can be selected from the combobox and saved with the OK button. Click ✕ to remove the property again later.

⊞ Delete Table; Click this icon to remove the table in which the cursor is currently positioned and all the table's contents.

11.6.10.2 Accept/Overwrite default selection of layout properties

The layout properties of a table cell are defined by the template developer using the style sheets. Depending on the configuration of the style sheets by the template developer, these layout properties can however be changed when editing the table in the DOM Editor. In this case the "Cell Properties" button in the top part of the DOM Editor is active:
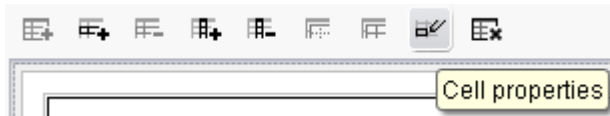
Figure 11-65: "Cell Properties" button within the DOM Editor

Click the icon to open a window with properties which can be assigned to the current cell:
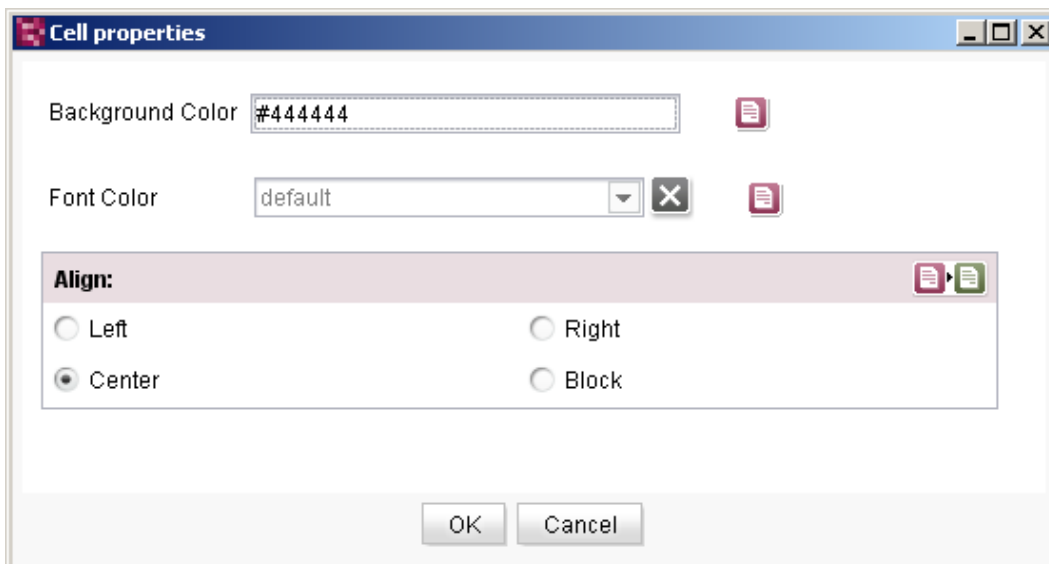
Figure 11-66: Example: Edit the properties of a table cell

Several cells to which the same property can be applied according to the template developer's specifications can be simultaneously selected and edited.

Values preset by the template developer are identified by pink coloured marking. You can manually overwrite these values or accept them for the respective cell by clicking the button. In both cases, the pink marking disappears. Click the button at any time to reset the value to the value preset by the developer.

The required properties can be selected from the input components and saved using

the [OK] button. Click ✖ to remove the property again later.

> ❗ *If invalid values are used for the background colour (e.g. #ZZZZZZ), the cell concerned is displayed with a white background colour in the relevant cell in the DOM Editor and in the subsequent output.*

11.6.10.3 Alternate continuous text and tables

To insert a **new text section in front of an inline table**, the cursor is positioned in the first cell of the required table and is moved to the left using the cursor key. The cursor is now displayed as follows (framed in red here):

Figure 11-67: Insert text section in front of inline table

The "Enter" key is then pressed to add a new text section in front of the table.

To insert a **new text section after an inline table**, the cursor is positioned in the last cell of the required table and is moved to the right using the cursor key. The cursor is now displayed as follows (framed in red here):

Figure 11-68: Insert text section after inline table

The "Enter" key is then pressed to add a new text section after the table.

11.6.10.4 Keyboard control of inline tables

The cursor keys ← → ↑ ↓ can be used to navigate the cells of inline tables (see also Chapter 11.11.1 page 389). Use the <TAB> key to move to the respective next cell. If the cursor is in the last cell of a row it goes to the first cell of the following row. At the end of a table, <TAB> is used to add a new row, provided the template developer's default values allow this. If no more rows may be added, the cursor remains in its current position. Use <SHIFT> + <TAB> to move the cursor backwards through the cells within the current table.

Press <RETURN> to insert new rows within a cell. Press the Back key to move through empty cells like with the cursor key ←.

For information on inserting text sections in front of and after inline tables, please refer to Chapter 11.6.10.3 page 355.

11.6.10.5 Further formatting options

The editor can use all format and link templates available in the DOM Editor for the text input, e.g. "bold", "underline", internal and external links. The insertion and editing of (nested) lists is also the same as in "simple" DOM Editors.

11.7　Working with the DOM Table input component

The DOM table is the input component for creating tables. Formatting can be defined both for highlighted text fragments and for whole cells. All functions of the icon bar, the context menus and the corresponding keyboard shortcuts are explained in this manual.
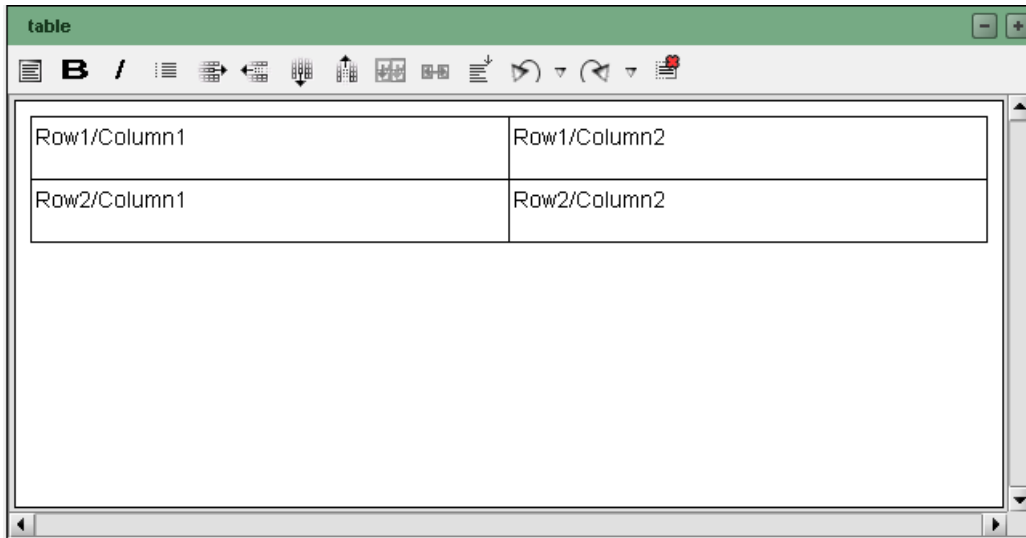
Figure 11-69: DOM_Table input component

> ! *Additional formatting options and the integration of tables into continuous text is provided by the so-called "inline tables" (from Version 4.1). For more information see Chapter 11.6.10 page 351.*

11.7.1 DOM Table: Icon bar

The standard functions which can be opened via the icon bar have already been described in the DOM Table chapter under Standard Input Components. See Chapter 10.12 page 278.

11.7.2 DOM Table context menus

11.7.2.1 Context menu in a table cell

Paste

Figure 11-70: DOM Table – Context menu in a table cell

Paste: Open this function to paste text fragments from the clipboard into the cursor's current position in the table cell.

11.7.2.2 Context menu on highlighted text
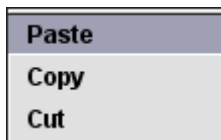
Paste
Copy
Cut

Figure 11-71: DOM Table – Context menu on highlighted text

Paste: Open this function to paste text fragments from the clipboard in the cursor's current position in the section.

Copy: Open this function to copy the selected text of the section into the clipboard.

Cut: This function is used to cut the selected text of the section and copy it into the clipboard.

11.7.2.3 Context menu on formatting

Delete format
Paste
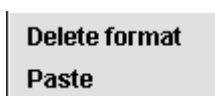
Figure 11-72: DOM Table – Context menu on formatting

Delete format: Open this function to remove the formatting for the selected text area. The text is retained.

Paste: Use this function to paste text from the clipboard at the position of the cursor in the cell.

11.7.2.4 Context menu on links

Figure 11-73: DOM Table – Context menu on links

Edit link: Open this function to open the input window of the link type used so that the settings made can be edited. For a detailed description of the settings for the various link types, see Chapter 10.23 page 291.

Remove Link: Open this function to remove the selected link. The link text is retained.

Paste: Use this function to paste text from the clipboard at the position of the cursor in the cell.

11.7.3 Checking spelling in the DOM Table

Checking spelling in the DOM Table input component is analogous to the spelling check in the DOM Editor (see Chapter 11.6.3 page 328).

11.7.4 Cell Properties
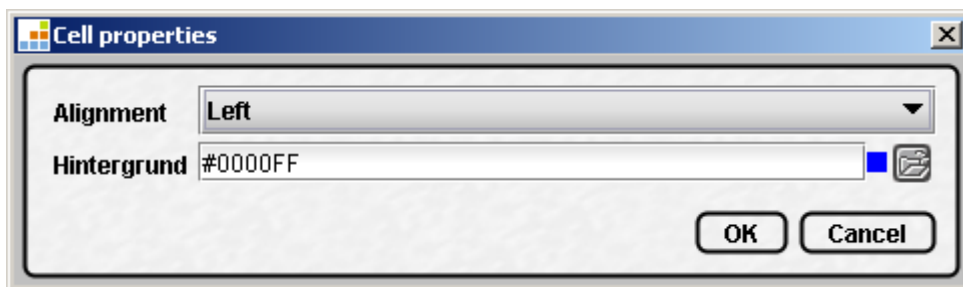
Figure 11-74: Cell Properties

Alignment: A combo box can be selected in this field to select the alignment of the text within the cell. The standard alignments are available: Left alignment, right alignment, centred and full justification.

Background: This field can be used to specify a background colour for the cell which should differ from the general background colour. This colour must be given as an HTML colour code in hexadecimal. However, a colour selection window is

available to provide support with the colour selection and can be opened using the
icon.
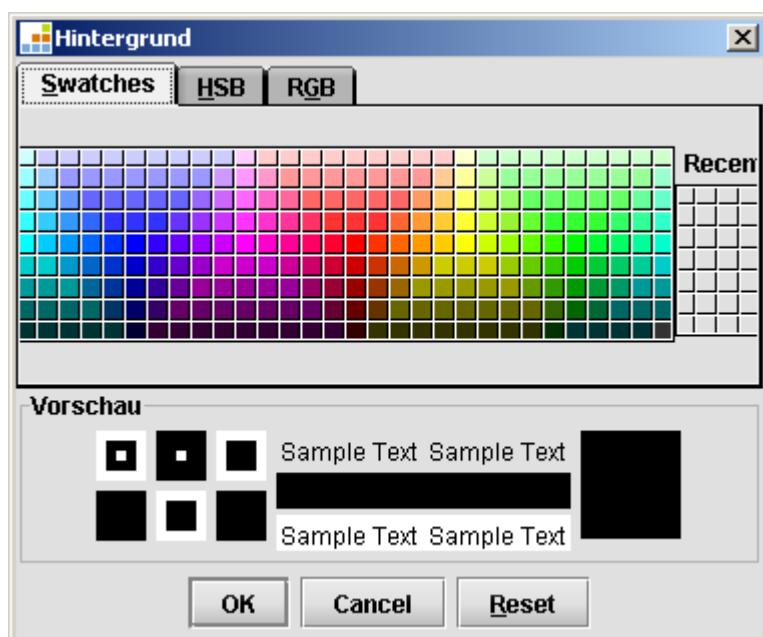
Figure 11-75: Select colour

The required colour can be precisely matched in this window with the help of the three tabs: Swatches, HSB and RGB.

11.8   Search in selection dialogs (from V4.1)

This function is available for FirstSpirit Version 4.1 and higher only. Screenshots are therefore displayed in the new "LightGray" Look & Feel. The search is not available in the "Classic" Look & Feel, the selection of objects in input components works as described in the relevant sub-chapters of 10 from page 249.

A search option is available within the selection dialogs in FirstSpirit JavaClient with which the various stores can be quickly searched through for objects which are to be referenced to, e.g. pictures and files from the Media Store or page references from the Site Store. This search option makes it easier to work with larger projects and extensive content.

When selecting objects using the "Open" icon, e.g. , in the input components

- CMS_INPUT_PICTURE   (Chapter 10.3 page 251),
- CMS_INPUT_FILE   (Chapter 10.10 page 273),
- CMS_INPUT_IMAGEMAP   (Chapter 10.13 page 281),
- CMS_INPUT_PAGEREF   (Chapter 10.19 page 288),
- FS_DATASET   (Chapter 10.25 page 305),
- FS_REFERENCE   (Chapter 10.27 page 311)
- and in links, whose configuration enables a reference to be selected (see also *FirstSpirit Manual for Developers*)   (Chapter 10.23 page 291),

a dialog opens which can be used to easily find and select the objects of the respective store. At the same time, only objects which

- may be selected on the basis of the input component default settings and
- which are visible to the user are displayed: Objects for which the user does not have any permissions are not displayed to them in the selection dialog, objects for which the user has "Viewable" permission are displayed to them only as hits in the results list of the Navigation tab. Here too, the new permissions configuration takes effect, as described in Chapter 13.1.4.3 page 454.

> ! *From FirstSpirit Version 4.2, depending on the template developer's settings, this selection dialog can provide advanced functions which are used to create Media Galleries. For further information on Media Galleries, please refer to Chapter 11.9 page 370.*

11.8.1   Navigation tab

On opening the dialog, the Navigation tab is active. It contains the Stores to choose from, as defined in the template of the respective selection component. If a reference to an object (a picture, a page, etc.) has already been selected and is now to be changed, this is displayed here. Otherwise the top level of the respective store is displayed.
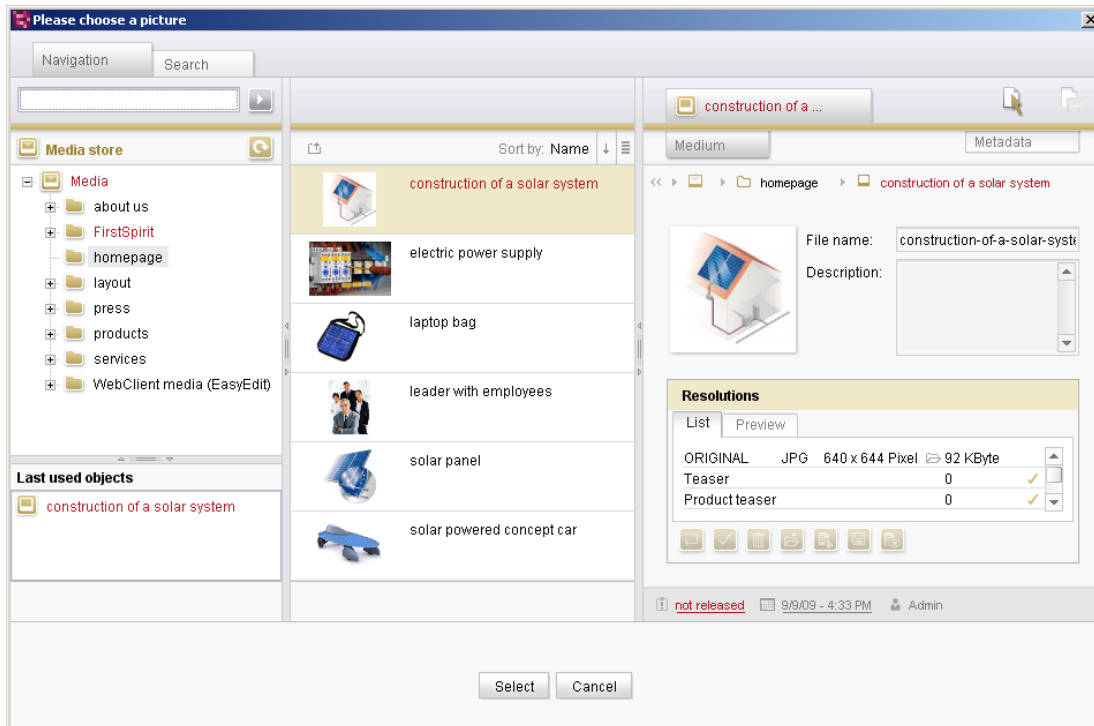
Figure 11-76: Picture selection – Navigation tab

11.8.1.1 Search field

The search field can be used to search all text fields of the respective store(s) with a full text search. The search term must be at least 3 characters long. Editorial content, e.g. texts in PDF files, is also included in the search. If the search is started with or the "Enter" key, the view of the dialog automatically switches to the Search tab (see chapter 11.8.2 page 365). If a folder was selected in the tree structure (see chapter 11.8.1.2 page 362), it is adopted as the starting point for the search.

11.8.1.2 Tree structure

The tree structure of the respective store can be seen in the left-hand column. If remote access to other FirstSpirit projects exists (see Chapter 11.14 page 416), this is indicated by the "(remote)" add-on. Here the user can navigate directly to a folder or to an object within the folder. Any folders of remote projects which exist are displayed in a separate area underneath the tree.

Search field: This search field can be used to search through all the text fields of the respective store by means of full text search. The search term must be at least 3 characters long. Editorial content is also included in the search, e.g. texts in PDF files. If the search is started with or the "Enter" key, the dialog view automatically

switches to the Search tab (see Chapter 11.8.2 page 365). If a folder has been selected, it is used as the starting point for the search.

Last used objects: Underneath the tree structure there is a list of the most recently used objects of the respective user; this makes it easier to find frequently used objects.

11.8.1.3 Results list

A view filtered by the folders of the respective store is possible in the middle column. Click a folder in the left-hand column and a list of the folders and objects which can be selected on the basis of the input component's default settings is displayed here, with display and reference names (see Chapter 3.1.4.2 page 69, "Display reference names in the tree" option) and thumbnails. Pictures are displayed with a reduced size preview of the picture, for all other file types the corresponding thumbnails of the file format are displayed (e.g. folder or PDF icons). The view can be sorted as follows:

Sort by: Click this area to open a list of criteria by which the results list can be sorted:
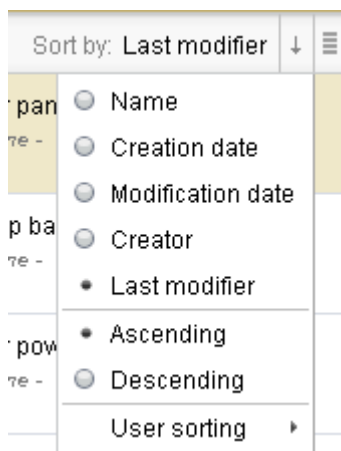
Figure 11-77: Definition of sort criteria

Name: Display and reference name of the object (see Chapter 3.1.4.2 page 69, "Display reference names in the tree" option)

Creation date: Date and time at which the object was created in JavaClient

Modification date: Date and time at which the object was last edited

Creator: Name of the user who created the object.

Last modifier: Name of the editor who most recently edited the object

Ascending: Ascending sort direction

Descending: Descending sort direction

User sorting: If sorting by creator or last editor, it is possible to select here whether the sort is based on the user's **Login**, **Name** or **Family name and First Name**.

The display is then updated according to the search criteria. If available, other information is also shown, e.g. the last modifier:
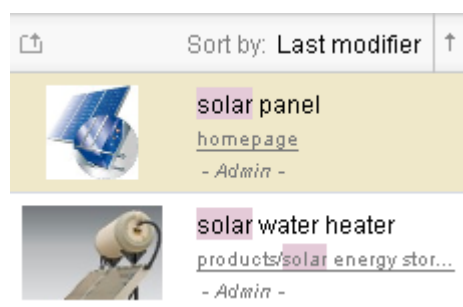
Figure 11-78: Sorting by "Last modifier"

The view of the results list can also be modified using the following icons:

⬆: this icon can be used to switch to the next higher folder.

↑: this icon can be used to set the sort direction (ascending, descending).

☰ ⊞: use these icons to switch between list view (arrangement below each other) and miniature view (arrangement next to each other).

11.8.1.4 Detailed view

At folder level: If a folder is selected in the middle column, all the folders and objects in the folder – provided they can be selected on the basis of the input component default settings – including all tabs are displayed. The settings which are also available in the respective store can be made for the selected folders (for folders of the Media Store see Chapter 6.5 from page 173, for menu levels in the Site Store, see Chapter 7.5 from page 228).

At object level: If an object is selected in the middle column, the details of the object including all tabs are displayed in the right-hand column. The settings which are also

available in the respective store can be made for the object (for media, see e.g. Chapter 6.6.1 from page 176, for files see Chapter 6.6.3 from page 188 and for page references, see Chapter 7.6 from page 235).

Editing and saving can be made using the familiar ![icon] and ![icon] icons of the icon bar. The ![Select] button is used to adopt the object displayed in the right-hand column in the input column and the dialog closes again.

11.8.2   Search tab

If you now switch to the Search tab, the stores defined by the template of the respective selection component can be searched through without having to navigate through the tree view. Search terms previously entered in the search field in the Navigation tab are retained when this switch is made.
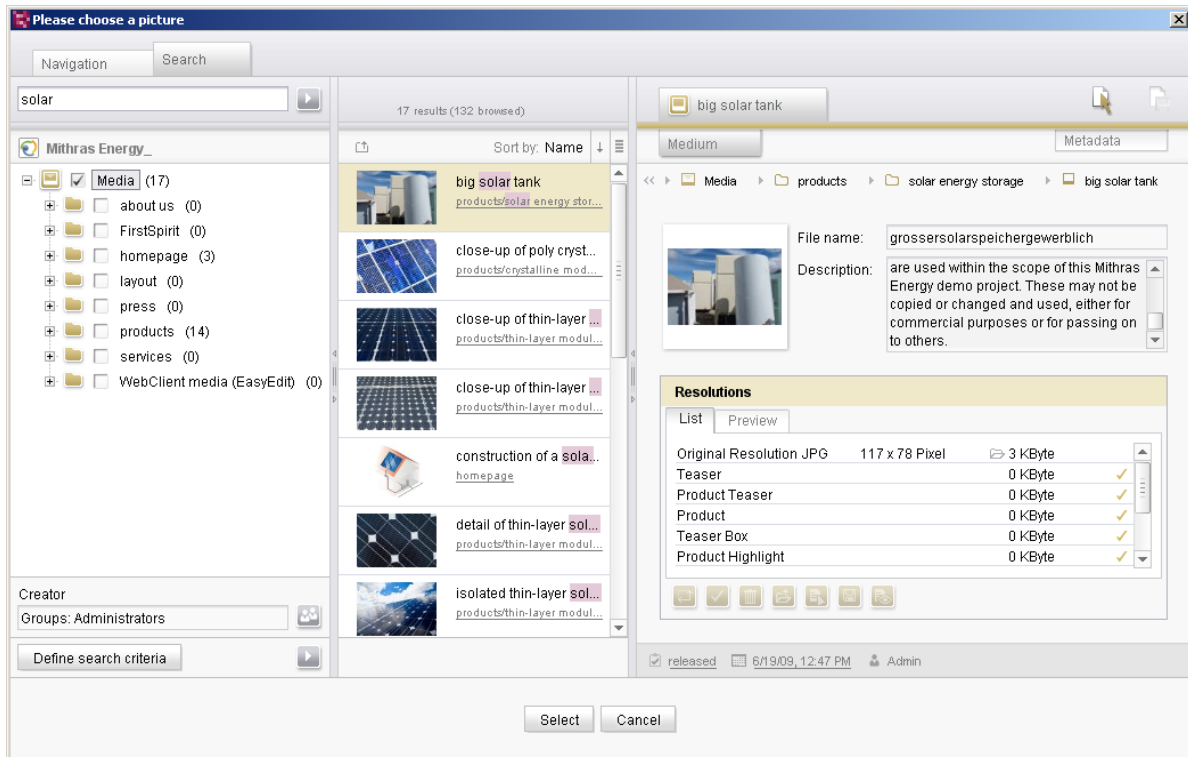
Figure 11-79: Picture selection – Search tab

Here too the view of the Navigation tab is divided into three parts (see Chapter 11.8.1 page 361). In addition to searching for a specific search term, search results can also be filtered (e.g. by modifier or change period) and sorted (see Chapter 11.8.2.2 page 369).

11.8.2.1 Tree structure

The tree structure of the store can be seen in the left-hand column. If remote access to other FirstSpirit projects exists (see Chapter 11.14 page 416), this is indicated by the "(remote)" add-on. The checkboxes can be used to limit the search to individual folders. Lower level folders are also included in the search and do not have to be additionally selected. To search, e.g. through the whole Media Store, the checkbox next to "Media" is ticked. After a search has been performed, the number of hits in the respective folder is displayed in brackets behind the folders.

Define search criteria: this button can be used to define filters with which the search can be limited. Ticks in the list indicate which criteria are already active. The ticks are disabled by clicking them again.
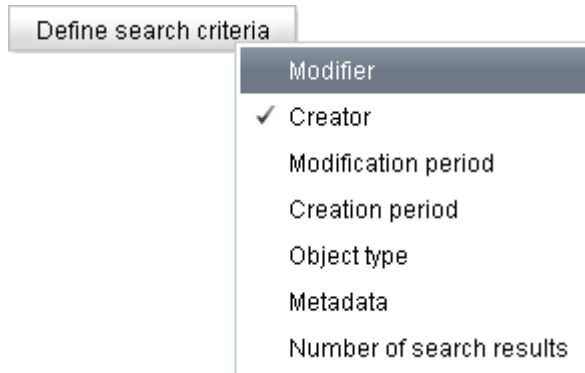
Figure 11-80: Define search criteria

Modifier: The icon can be used to limit the search to objects most recently edited by certain users or groups (for details of allocation of users and groups, see Chapter 13.2.4 page 467).

Creator: The icon can be used to limit the search to objects created by certain users or groups (for details of allocation of users and groups, see Chapter 13.2.4 page 467).

Modification period: This combobox can be used to limit the search to objects changed within a certain period. If the "user defined" option is selected, the calendar icons can be used to individually define the period.

Creation period: This combobox can be used to limit the search to new objects created within a certain period. If the "user defined" option is selected, the calendar icons can be used to individually define the period.

> *Filtering by the search criteria*
> *- Modifier*
> *- Creator*
> *- Modification period*
> *- Creation period*
> *requires a new data structure in the search index introduced with FirstSpirit Version 4.1. As this information is missing in projects migrated from Version 4.0 to 4.1, a corresponding search does not return any hits. Each time changes are made with Version 4.1, the user/modifier is included in the index and is found during the search.*

Object type: This combobox can be used to limit the search to certain types of objects (pictures, files, pages).

Metadata: If working with metadata has been configured for a project, this filter can be used to limit the search to objects for which specific metadata are defined. Another dialog opens, in which criteria and values for can be defined for the metadata search:
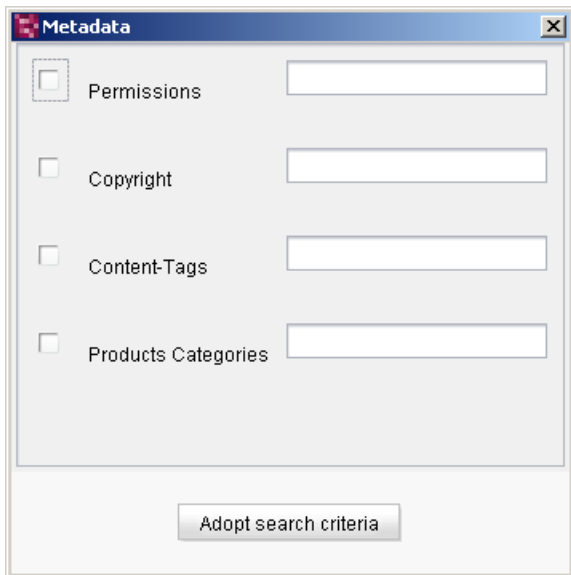
Figure 11-81: Metadata search example

The metadata input components defined for the project are displayed here. The respective checkbox must be enabled to search for specific metadata types (here "Permissions", "Copyright", etc.). In addition, it is possible to specify the specific value to be searched for, for each type. Click **Adopt search criteria** to copy the criteria into the search and to close the dialog again.

The criteria defined in the dialog in Figure 11-81 are displayed in the search criteria area as follows:
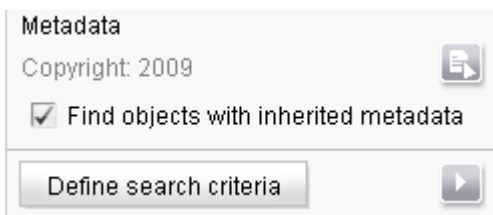
Figure 11-82: "Metadata" search filter

Here the **Find objects with inherited metadata** option can be used to also define whether objects whose metadata was only inherited are also to be included in the search. If a checkbox is not ticked, only search results which explicitly contain the

value given for the search criterion in the meta data are taken into account.

The Edit icon can be used to change the criteria and values of the metadata search filter.

For further information on using metadata, see also Chapter 11.3 page 317 and the *FirstSpirit Manual for Developers.*

Number of search results: The selection list can be used to limit the number of search results. The "User defined" option enables the number to be individually defined. If the number of hits exceeds the number of search results defined here, a corresponding message is issued during the search:
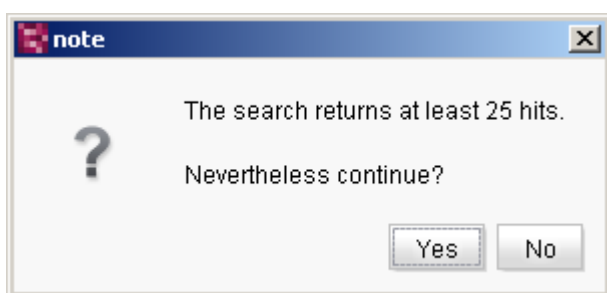
Figure 11-83: Number of search results

Click **Yes** and the search is continued, all hits are subsequently displayed in the results list. Click **No** and the search is cancelled and only the hits found up until then are displayed.

The **Search field** above the tree structure can be used to search through all the text fields of the respective store by means of full text search. The search term must be at least 3 characters long. The search is started using or with the "Enter" key. Alternatively, the search can also be started using next to the **Define search criteria** button. Editorial content is also included in the search, e.g. texts in PDF files.

11.8.2.2 Results list

The results of the search are displayed in the middle column, analogous to those in the Navigation tab (Chapter 11.8.1.3 page 363). During the search process the progress is displayed above the results list; after the search has finished, the total number of hits is displayed. The search term is highlighted in colour in the results list and the path is also displayed. The sort options are also analogous to those in the Navigation tab.

11.8.2.3 Detailed view

The details of the object selected in the results list including all tabs are displayed in the right-hand column. The settings which are also available in the respective store can be made for the object (for media, see e.g. Chapter 6.6.1 from page 176, for files see Chapter 6.6.3 from page 188 and for page references, see Chapter 7.6 from page 235).

Editing and saving can be made using the familiar [icon] and [icon] icons of the icon bar. The [Select] button is used to copy the object into the input component and then the dialog closes again.

11.8.3 Navigation using the keyboard

Apart from the keyboard shortcuts described in Chapter 11.11 from page 387, it is also possible to navigate through the selection dialogs as follows:

<TAB> (+ <Shift>)	=	Switch between the tree structure, the results list, the "Select" /"Cancel" button and the search field
<ENTER>	=	on folders in the results list: Display the folder contents in the results list
	=	on elements in the results list: Display the next element or display the element in the detailed view
<BACKSPACE>	=	in the results list: Display the next higher level
Double-click	=	on folders in the tree structure: Open the next folder level
	=	on folders in the results list: Display the folder contents in the results list
	=	on elements in the results list: Copy the element into the input component; the selection dialog closes

11.9 Working with Media Galleries (from V4.1)

From Version 4.1, FirstSpirit provides an option for selecting pictures from the Media Store and displaying them in a gallery view. The same picture can be used in several galleries, each with different description texts.

> ❗ *The configuration of media galleries depends on the settings of the template developer and can vary from project to project. It is therefore possible that the illustrations shown here differ from the view in the respective project or that less functions are available in the respective project than described in this document.*

The following options are provided for outputting galleries on a website:

- **Individual galleries:** These mostly consist of a entry page and a page with the specific gallery. The entry page shows the general information on the gallery such as its name and a description and one or several teaser images. The gallery can be used in the middle of the main area of the page or, for example, in teaser columns, in addition to other information on the page. The elements of the gallery are linked and lead to a page on which the pictures of the gallery can be paged through in an order definable by the editor (see also Chapter 11.9.4 page 380).
- **Gallery overview/s**: If several galleries exist in a project, these can be output on a page or on several pages of a page group (see Chapter 7.5.3 page 232). Depending on the project settings, they can also be filtered by categories and output in different sort configurations (e.g. by creation date, gallery title) (see also Chapter 11.9.5 page 384).

The creation and maintenance of galleries requires several steps in different stores. These are explained in the following.

The following stores are relevant for the editorial maintenance of media galleries:

- **Media Store:** The pictures to be subsequently displayed in a gallery are stored here (see Chapter 11.9.1 page 372).
- **Content Store:** All information on galleries is managed in corresponding content (data source) (in the following *Gallery data source*). Here, e.g. media from the Media Store are assigned to individual galleries and teaser images and description texts are deposited. In addition, galleries can be assigned to categories (see Chapter 11.9.2 page 372).
- **Page Store:** Within the Page Store, individual galleries can be integrated on a page (see Chapter 11.9.4 page 380) or gallery overviews can be generated (see Chapter 11.9.5 page 384). In addition, the Page Store can also be used to maintain gallery data (see Chapter 11.9.3 page 380).

11.9.1 Upload and manage media for the gallery

Pictures which are to be subsequently displayed in a gallery are uploaded into the Media Store first, as described in Chapter 6 from page 160, and there they can be maintained as usual. Depending on the template developer's default settings, folders with several media can also be used. Descriptions of the media can be deposited in the Content Store in the next step (see Chapter 11.9.2 page 372).

11.9.2 Creating and maintaining galleries via the Content Store

A separate data set for each gallery must be created in the Gallery data source.

> **!** *The respective name of the content in the Content Store depends on the template developer's default settings.*

> **!** *For detailed information on working with content, see Chapter 5 page 133.*

A gallery content data set contains at least one input component of the type "Data record selection list" (CMS_INPUT_CONTENTLIST, see Chapter 10.6 page 263), via which the media for the respective gallery can be selected. In addition, depending on the configuration of the template developer, other gallery data can be deposited, which is used for the entry page, e.g. name and description of the gallery and teaser images. As a default, this information can be subsequently output with each use of the gallery in the Page Store (see Chapter 11.9.4 page 380).

Figure 11-84 shows an exemplary input screen form for a gallery data set:

Figure 11-84: Gallery data set example

Category: Depending on the template developer's settings, the current gallery data set can be combined with data from other content. The "Category" combobox can be used in Figure 11-84 for example to select a category for the gallery. This can be used lager for filtered output (see Chapter 11.9.5 page 384).

Teaser-Picture: If a teaser image from the gallery is to be used, this can be dragged from the media list described in Chapter 11.9.2.1 page 375 (see also Figure 11-87) and into this input component by means of Drag & Drop.

Media: Media for the gallery can be selected using the "Add" icon by means of the selection dialog (see Chapter 11.9.2.1 page 375) or 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 11.4 page 318). In addition, the option for multiple selection is available in 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.

Detailed information on working with the input components which can be used in addition to these is given in Chapter 10 from page 249.

11.9.2.1 Select media for galleries

The "Data set selection list" input component (labelled "media" in Figure 11-84) can be used to select media for the gallery.

Add: When adding elements to the input component, the dialog for multiple selection of media opens:
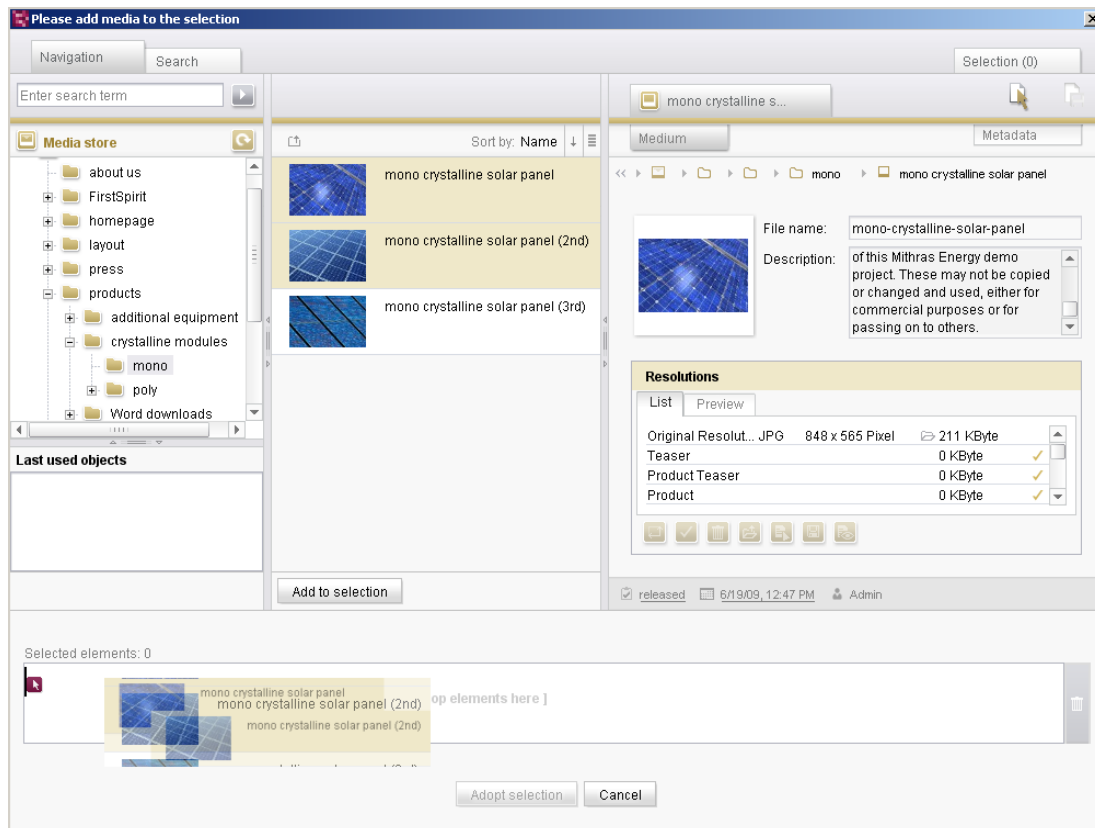
Figure 11-85: Adding pictures to the media gallery (selection dialog)

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. Depending on the template developer's default settings, folders can also be selected.

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 11-86).

> ![!] *Detailed information on searching for media using this selection dialog, see Chapter 11.8 page 360.*

Add to selection   The required elements are selected using the "Add to the selection" button. Multiple selection is possible by pressing the <CTRL> and <SHIFT> key. Each element can only be selected once.

Selection (2)   The number of selected elements is displayed in brackets on the "Selection" tab. A folder only counts as one element. Click the tab to open a list of the selected elements:
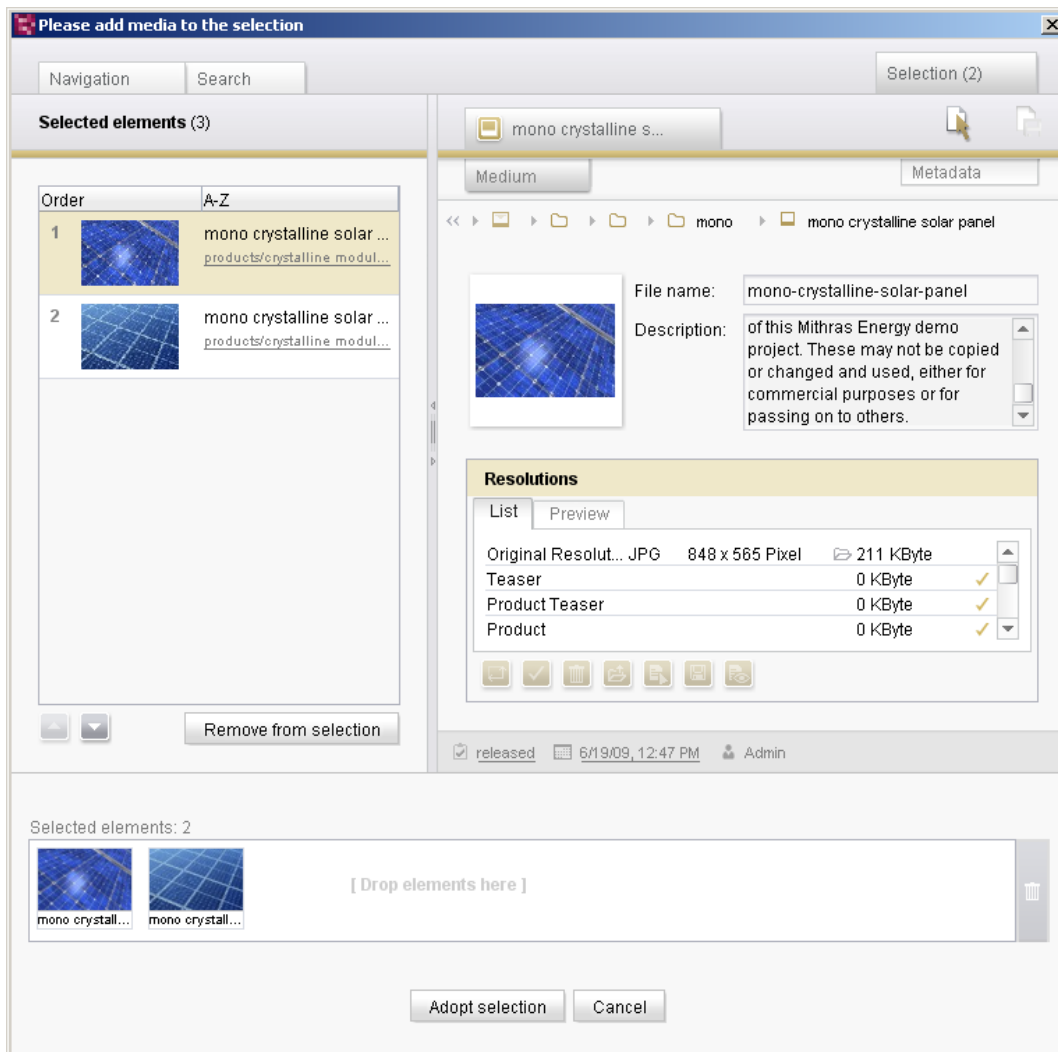
Figure 11-86: Selection tab: Edit selected media

Selected elements: The sum of the selected media including media contained in folders is displayed here. Folders are not counted.

List: The selected media and folders are displayed on the left-hand side in the order in which they were added. They are also subsequently output in the gallery in the order of the corresponding ordinal numbers. Click the heading fields to sort the list by the values in the corresponding column: either by order (takes into account the date/time the media/folders were added to the selection) or alphabetically (by reference or display names).

Up / Down: click these icons to change the ordinal number of the selected medium. This affects the order of the output.

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

Remove from selection This button can be used to remove selected media from the list. Several media can be selected by simultaneously pressing the <Ctrl> and <Shift> key or by clicking on a medium and dragging the mouse. with the mouse key pressed, to media above or below the first medium.

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:

> *If necessary, media can also be removed at any time using the input screen form for the gallery data set (see Figure 11-87) and the order can be changed.*

Detailed view: The detailed view of the selected medium or the content of folders is displayed on the right-hand side. By clicking on the preview icon for individual media, a view of the picture can be requested in the respective selected resolution.

The list of selected media can be exited again by clicking the "Navigation" or "Search" tab and other media can be selected.

If all the required media for the gallery have been selected and are included in the list, the media can be copied into the gallery using the Adopt selection button; if folders have been selected, all the media in them is "unpacked" (extracted).

> ![!] *Depending on the project settings, the number of media which can be selected for each gallery can be limited. If the allowed number is exceeded the following message is displayed: "Maximum [maximum allowed number of media] entries may be selected, current selection: [number of media selected]. The surplus entries will not be accepted". Only the allowed number is accepted.*

The input screen form can then look like the following:
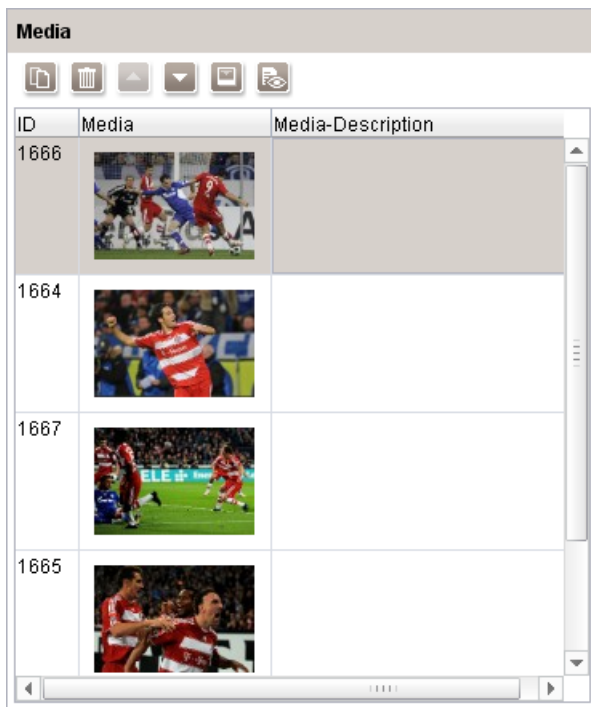
Figure 11-87: Media gallery: List of the selected media

The selected media are displayed in the input screen form in list form with ID. The media description is initially empty and is not entered until the next step (see Chapter 11.9.2.2 page 379).

Delete Section: click this icon to remove the selected medium from the gallery.

Move Section Up/Down: click this icon to move the selected medium one position up or down in the order. Alternatively, media can also be moved by means of Drag & Drop.

11.9.2.2   Deposit information on media

Display (or double-click): click this icon to open an input screen form for the respective medium:

Figure 11-88: Media data set example

Information, e.g. description texts on the selected media, can be entered here. If necessary, the picture can also be exchanged here. Changes are accepted using the "Save" icon.

With FirstSpirit Version 4.2, a convenient option was created for series editing within the CMS_INPUT_CONTENTLIST input component. The icons in the top area of the media data set can be used to directly switch to the first / previous / next / last data set of the input component. If several consecutive entries within the input components are to be changed, the detailed dialog does not have to be closed after each change and then re-opened on a new data set.

> **!** *Detailed information on working with the input components which can be used in addition to these is given in Chapter 10 from page 249.*

11.9.3 Creating and maintaining galleries using the Page Store

As an alternative to maintaining gallery data through the Content Store (see Chapter 11.9.2 page 372), this data can also be maintained in the Page Store via sections with the input component of the type "Select/Edit Data Record" (CMS_INPUT_OBJECTCHOOSER, see Chapter 10.7 page 265). This is done using the "New" and "Edit" buttons.

> **!** *For further information on creating and editing gallery data via the Page Store, see Chapter 11.9.4 page 380.*

11.9.4 Using individual galleries

To insert an individual gallery into a page in the Page Store, a corresponding section template must be selected.

> **!** *The respective name of the section template for media galleries depends on the settings of the template developer.*

> **!** *Figure 11-89 shows an example of a section with a media gallery. Depending on the template developer's settings, the input components used in the project can differ from those shown here. However, a section with media galleries contains at least one input component of the type "Select/Edit data record" (CMS_INPUT_OBJECTCHOOSER, see Chapter 10.7 page 265).*

Figure 11-89: Example of Section tab with section-dependent settings

This tab is used to maintain the gallery data for this section, which is subsequently displayed as the gallery entry page in the output (see Figure 11-91). In this example, the input components can be used to give the gallery name, a gallery description and teaser images.

> ❗ *The required gallery including the media are selected in the next step using the "Gallery" tab (Figure 11-90). To output the entry page data deposited for this gallery in the* **Gallery data source** *(see Chapter 11.9.2 page 372), the fields of this tab must remain unfilled.*

> ❗ *Detailed information on working with the input components which can be used in addition to these is given in Chapter 10 from page 249.*

Figure 11-90: Example of Section – tab with CMS_INPUT_OBJECTCHOOSER

This tab is used to select the gallery for the section. If necessary, an existing selection can be undone or a new gallery can be created.

[Reset] This button can be used to undo an existing selection of a data set.

[New] This button can be used to create new data sets in the gallery content (see Chapter 11.9.2 page 372). Click to open the input screen form from Figure 11-84. The data entered is saved as a new data set in the content and is copied into the current section.

[Select] This button can be used to select a data set from the Gallery data source.

Click to open a dialog via which the required data set can be selected from the content. Press "OK" to copy it into the section.

Edit   This button can be used to edit an already selected data set. Click to open the input screen form from Figure 11-84. The changed data is saved in the content and is copied into the current section.

> ! *The data displayed on this tab is data from the Gallery data source. If data is also entered on the tab with the section-dependent settings (Figure 11-89), the section-dependent data is output instead of the values from the gallery data set selected here.*

> ! *With FirstSpirit Version 4.2 the input component CMS_INPUT_OBJECTCHOOSER has been enhanced and modiified. For further information on working with this input component see Chapter 10.7 page 265.*

The section may subsequently look like this:

Sports Gallery

The best moments of the season's top game FC Bayern Munich contra FC Schalke 04

Galerie (0 Pictures) - Category: Sport - Date: 29.09.2009

Figure 11-91: Exemplary output of gallery entry page

In this example, the gallery entry page is displayed with a title, description and a teaser image. Click these elements to switch to a page with the actual gallery:

Sports Gallery
The best moments of the season's top game FC Bayern Munich contra FC Schalke 04

To article   < Picture 1 from 7 >   All Galleries

Gallery (7 Pictures) - Category: Sports - Date: 29.09.2009

FC Bayern - Picture 1

Figure 11-92: Exemplary output of media gallery

The first medium of the gallery is displayed here with the respective data. The navigation can be used to page through the gallery media, namely in the order set in Chapter 11.9.2.1 (from page 375).

11.9.5 Using gallery overviews

Sections for the gallery overviews are inserted in pages of the Page Store using special templates. This is usually the task of the template developer. Sections for holding editorial content can be inserted as usual above or below these sections.

> ! *Gallery overviews are sorted and filtered in the Site Store by the template developer.*

An overview of all galleries can, for example, look like this:

Figure 11-93: Exemplary overview of all galleries

All the available galleries (framed in red in the Figure) are displayed here with names, description texts and teaser images. Click these elements to switch to the page with the actual gallery (cf. Figure 11-92).

If categories have been defined by the template developer and assigned during maintenance (see Figure 11-84), galleries can also be output filtered by these categories. A gallery overview filtered by the "Sport" category can, for example, look like this:
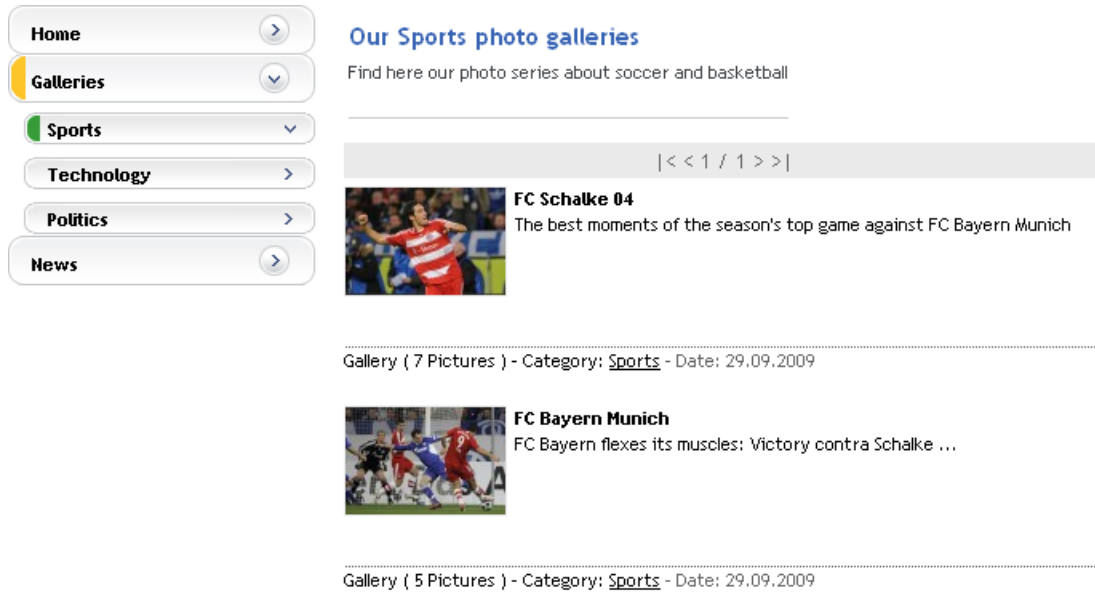
Figure 11-94: Exemplary overview of the "Sport" category galleries

The "sport" category galleries are displayed here with names, description texts and teaser images. Click this element to switch to a page with the actual gallery (cf. Figure 11-92).

11.10 Accept fallback values

The "Accept Fallback Values" icon can be seen in the top right-hand corner of the input component of the Page Store. This icon always appears if the project developer has specified a default value for this input component.
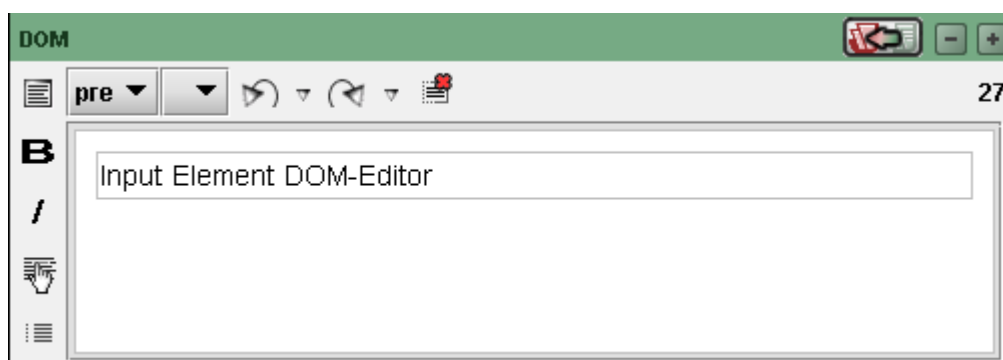
Figure 11-95: Input component with the "Accept Default Values" icon

 Click this icon to reset the content of the input component back to the fallback value specified by the project developer.

A confirmation prompt appears asking whether you are sure you want to delete the current content.

Confirm the prompt with OK to delete the current content of the input component and replace it with the fallback value.

The input component is surrounded with a different-coloured frame to indicate that its content is a fallback value.
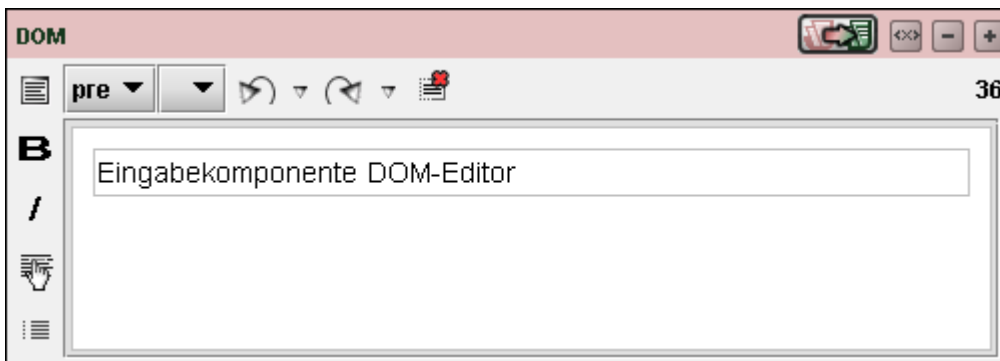
Figure 11-96: Input component with fallback value

11.11 Keyboard shortcuts

It is of course also possible to shorten frequent work with appropriate keyboard inputs in FirstSpirit. The cursor keys can be used to move in the tree structure or to switch between the tabs. The next level can be opened with cursor right or Return; cursor left closes the level again or goes to the next higher level. The Tabulator key can be used to switch from the left-hand side of the screen to the right-hand side of the screen and to work through the elements in the tabs. Use SHIFT + Tabulator key to move back through the fields. Use the CTRL key to open combo boxes and the arrow keys to navigate through them.

The following keyboard shortcuts can be used instead of the corresponding buttons in the client's toolbar, the respective entries from the context menu or the relevant mouse actions (CTRL+S means that the CTRL key and the "S" key have to be

pressed simultaneously. F5 refers to the relevant function key):

Buttons of the client's toolbar:

ALT + ← / →	**=**	**Go back / forwards to objects which have already been selected once during the current session**
F5	**=**	**Update**
CTRL + E	**=**	**Edit mode on / off**
CTRL + S	**=**	**Save**
CTRL + P	**=**	**Preview button**
CTRL + N	**=**	**New**
Del	**=**	**Delete**
CTRL + T	**=**	**Task list on/off**
F1	**=**	**Open Online help**

Context menu items

Application key 📑	**=**	**Open the context menu of the respective node**
CTRL+ SHIFT + N	**=**	**New folder**
CTRL+ Z **or CTRL+ SHIFT + Z**	**=**	**Undo (back to the last saved status of the node, with confirmation dialog)**
CTRL + X	**=**	**Cut**
CTRL + C	**=**	**Copy**
CTRL + V	**=**	**Insert**
CTRL + D	**=**	**Create document group (Site Store only)**
F9	**=**	**Rename**
CTRL + U	**=**	**Display usages (if only one use exists the focus switches to this use)**
CTRL + SHIFT + E	**=**	**Cancel editing, data which are not saved will get lost**
CTRL + H	**=**	**Open version history (not on data records in the Content Store)**
CTRL + W	**=**	**Close the actual workspace**
<User-defined>	**=**	**Depending on the project configuration user-defined keyboard shortcuts can be at the editor's disposal for starting a workflow and switching it to another state (Chapter 12 page 422).**

Selecting and moving objects

CTRL + mouse click	**=**	**Multiple selection of single elements**
SHIFT + mouse click	**=**	**Multiple selection of several elements**
SHIFT + Cursor	**=**	**Multiple selection of several elements / Extend a selection upwards / downwards**
CTRL + A	**=**	**Select all elements**
Drawing with mouse pointer	**=**	**Move objects**

CTRL + drawing with mouse pointer	=	**Copy objects**

Other functions

CTRL + Z	=	**Undo last operation**
CTRL + F	=	**Open full-text search**
F3	=	**Find next**
Shift + F3	=	**Find previous**
ALT + P	=	**Display node information**
F10	=	**Activate menu bar, navigate per cursor keys**
F11	=	**Full screen mode for the current window to enable / disable**
F12	=	**Open the special characters table**
SHIFT + CTRL + drawing with mouse pointer	=	**Create section reference (Page Store only)**
CTRL + L	=	**Show the line numbering when editing a template (Template Store only)**

11.11.1 Keyboard shortcuts in the DOM Editor

The following keyboard shortcuts apply in the DOM Editor:

<ENTER>	=	Insert a new paragraph
←	=	Move cursor one position to the left
→	=	Move cursor one position to the right
↑	=	Move cursor up by one position
↓	=	Move cursor down by one position
<CTRL> + ←	=	Go to the next leading space character
<CTRL> + →	=	Go to the next following space character
<HOME>	=	Go to the first character in the current row
<END>	=	Go to the last character in the current row
<ALT> + <HOME>	=	Go to the first character of the section
<ALT> + <END>	=	Go to the last character of the section
<CTRL> + <HOME>	=	Go to the first character in the DOM Editor
<CTRL> + <END>	=	Go to the last character in the DOM Editor
<SHIFT> + ←	=	Highlights the text from the current position up to the preceding character
<SHIFT> + →	=	Highlights the text from the current position up to the following character
<SHIFT> + ↑	=	Highlights the text from the current position up to this position in the row above
<SHIFT> + ↓	=	Highlights the text from the current position up to this

		position in the row beneath
<SHIFT> + <HOME>	=	Highlights the text from the current position up to the first character in the current row
<SHIFT>+ <END>	=	Highlights the text from the current position up to the last character in the current row
<SHIFT> + <CTRL> + ←	=	Highlights the text from the current position up to the next leading space character
<SHIFT> + <CTRL> + →	=	Highlights the text from the current position up to the next following space character
<SHIFT> + <ALT> + <HOME>	=	Highlights the text from the current position up to the first character of the section
<SHIFT> + <ALT> + <END>	=	Highlights the text from the current position up to the last character of the section
<SHIFT> + <CTRL> + <HOME>	=	Highlights the text from the current position up to the first character in the DOM Editor
<SHIFT> + <CTRL> + <END>	=	Highlights the text from the current position up to the last character in the DOM Editor
<CTRL> + <A>	=	Highlights the whole content of the DOM Editor
Double-click	=	Select the actual word (incl. punctuation marks)
Triple-click	=	Select the current line
Quadruple-click	=	Select the current section
<CTRL> + <C>	=	Copies the highlighted content into the clipboard
<CTRL> + <X>	=	Cuts the highlighted content and copies it into the clipboard
<CTRL> + <Del>	=	Delete a word from the cursor position to the end of a the word, following punctuation marks included
<CTRL> + <Backspace>	=	Delete a word from the cursor position to the beginning of the word
<CTRL> + <V>	=	Paste the content of the clipboard at the cursor's position
<CTRL> + <F>	=	Opens a window for find text in the DOM Editor
<CTRL> + <R>	=	Opens a window to find and replace text in the DOM Editor
<F3>	=	Goes to the next search result
<SHIFT> + <F3>	=	Goes to the previous search result
<CTRL> + <Z>	=	Undoes the last change
<CTRL> + <SHIFT> + <Z>	=	Restore changes which have been undone (redo)
<CTRL> + <Y>	=	Restore changes which have been undone (redo)
<Screen ↑>	=	Page up without moving the cursor
<Screen ↓>	=	Page down without moving the cursor

For further information about the keyboard shortcuts that can be used in tables in DOM Editors from FirstSpirit version 4.1 see Chapter 11.6.10.4

11.11.2 Keyboard Shortcuts DOM Table

<ENTER>   = Insert line break at the current position

← = Move cursor one position to the left within a cell or to the next cell at the left

→ = Move cursor one position to the right within a cell or to the next cell at the right

↑ = Move cursor one position up within a cell or one cell up

↓ = Move cursor one position down within a cell or one cell down

<CTRL> + ← = Go to the next preceding space character

<CTRL> + → = Go to the next following space character

<HOME> = Go to the first character in the current cell row

<END> = Go to the last character in the current cell row

<ALT> + <HOME> = Go to the first character of the cell

<ALT> + <END> = Go to the last character of the cell

<SHIFT> + ← = Highlights the text from the current position up to the preceding character.
Each additional click on ← the selection will expand one more character to the left. If the cursor is in front of the 1. character of a cell clicking ← will select the actual cell as well as the next cell to the left. If there is not any cell left to the actual cell the actual cell as well as the next cell above will be selected.

<SHIFT> + → = Highlights the text from the current position up to the next character.
Each additional click on → the selection will expand the one more character to the right. If the cursor is behind the last character of a cell clicking → will select the actual cell as well as the next cell to the right. If there is not any cell right to the actual cell the actual cell as well as the next cell under it will be selected.

<SHIFT> + ↑ = Highlights the text from the current position up to the same position in the row above.
Each additional click on ↑ the selection will be expanded one line above. If the cursor is in the first line of a cell clicking ↑ the actual cell and the cell above will be selected.

<SHIFT> + ↓ = Highlights the text from the current position up to the same position in the following row.
Each additional click on ↓ the selection will be expanded one line beneath. If the cursor is in the last line of a cell clicking ↓ the actual cell and the cell under it will be

selected.

<SHIFT> + <HOME>	=	Highlights the text from the current position up to the first character in the current cell row
<SHIFT>+ <END>	=	Highlights the text from the current position up to the last character in the current cell row
<SHIFT> + <CTRL> + ←	=	Highlights the text from the current position up to the next preceding space character
<SHIFT> + <CTRL> + →	=	Highlights the text from the current position up to the next following space character
<SHIFT> + <ALT> + <HOME>	=	Highlights the text from the current position up to the first character of the cell
<SHIFT> + <ALT> + <END>	=	Highlights the text from the current position up to the last character of the cell
<CTRL> + <A>	=	Highlight the whole content of the DOM-Table
Double-click	=	Select the current word (incl. punctuation marks)
Triple-click	=	Select the current line of a row
Quadruple-click	=	Select the current cell
<CTRL> + <C>	=	Copies the highlighted content into the clipboard
<CTRL> + <X>	=	Cuts the highlighted content and copies it into the clipboard
<CTRL> + <Backspace>	=	Delete a word from the cursor position to the beginning of the word
<CTRL> + <Del>	=	Delete a word from the cursor position to the end of a the word, following punctuation marks included
<CTRL> + <V>	=	Paste the content of the clipboard at the cursor's position
<CTRL> + <F>	=	Open a window for finding text in the DOM-Table
<CTRL> + <R>	=	Open a window for finding and replacing text in the DOM-Table
<F3>	=	Goes to the next search result
<SHIFT> + <F3>	=	Goes to the previous search result
<CTRL> + <Z>	=	Undoes the last change
<CTRL> + <SHIFT> + <Z>	=	Restore changes which have been undone (redo)
<CTRL> + <Y>	=	Restore changes which have been undone (redo)
<Screen ↑>	=	Page up without moving the cursor
<Screen ↓>	=	Page down without moving the cursor
<TAB>	=	Cursor goes to the next cell. If the cursor is in the last cell of a row it goes to the first cell of the following row. If the cursor is in the last cell of the table a new row will be inserted below the current one.
<SHIFT> + <TAB>	=	Cursor goes to the previous cell. If the cursor is in the first cell of a row it goes to the last cell of the row above.

11.12 Version Management in the FirstSpirit JavaClient

A version history exists for all project data in a FirstSpirit project which shows how the data has been changed over time. The primary objective is the most continuous possible traceability of all changes and the possibility of resetting these at any time.

11.12.1 Terms and concepts

11.12.1.1 Versioning and historization

First, let us consider the basic objects of a Content Management System, for example a medium or an individual page or a section. A new version of this object is made for each change made to such an object by the editor. Thus, an object has a version history on the base of which it is possible to trace which changes were made by which persons over time.

The version history of the individual objects is not sufficient to ensure complete traceability of all changes as the individual basic objects are grouped together within the Content Management System to form more complex structures. In FirstSpirit, e.g. pages are compiled from individual sections and are intertwined in the Site Store to form navigation. Changes to these structural aspects must therefore also be part of the versioning. The versioning of the basic objects and the structural aspects give a versioned description of the whole system status which enables changes to be traced.

A further aspect which must also be taken into account within the scope of the versioning is the implementation of procedures for the approval and release of changes. A release procedure is usually implemented via an appropriately professional workflow (see Chapter 12.3 page 425). At the technical level, when a change is released a specific version of an object is labelled as being "released"..

11.12.1.2 Repositories and revisions

All the information required is stored in a FirstSpirit Repository, a central place in which the data structures (media, pages, templates, etc.) required by the Content Management System are managed. Each FirstSpirit project has its own self-contained repository. A special technique is used in the FirstSpirit Repository to manage the change in data over time: The so-called Revision Management.

A revision can be thought of as a kind of "snapshot" of the whole repository at a specific point in time. Unlike a version which usually only relates to a single object, the complete state of all objects is described in a revision.

Revisions are described by consecutive numbering, whereby there is always precisely one current revision for the whole repository. When a repository is edited, all changes made in a logical context (how ever it is defined) are linked with a new revision number. The revision number results from the last current revision number of the whole repository increased by one. All unchanged objects keep their old revision numbers. If an object is changed it is not overwritten in the repository but instead is inserted as a new object (with a higher revision number).

A version history of individual objects can be opened at defined objects of a project using the context menu.
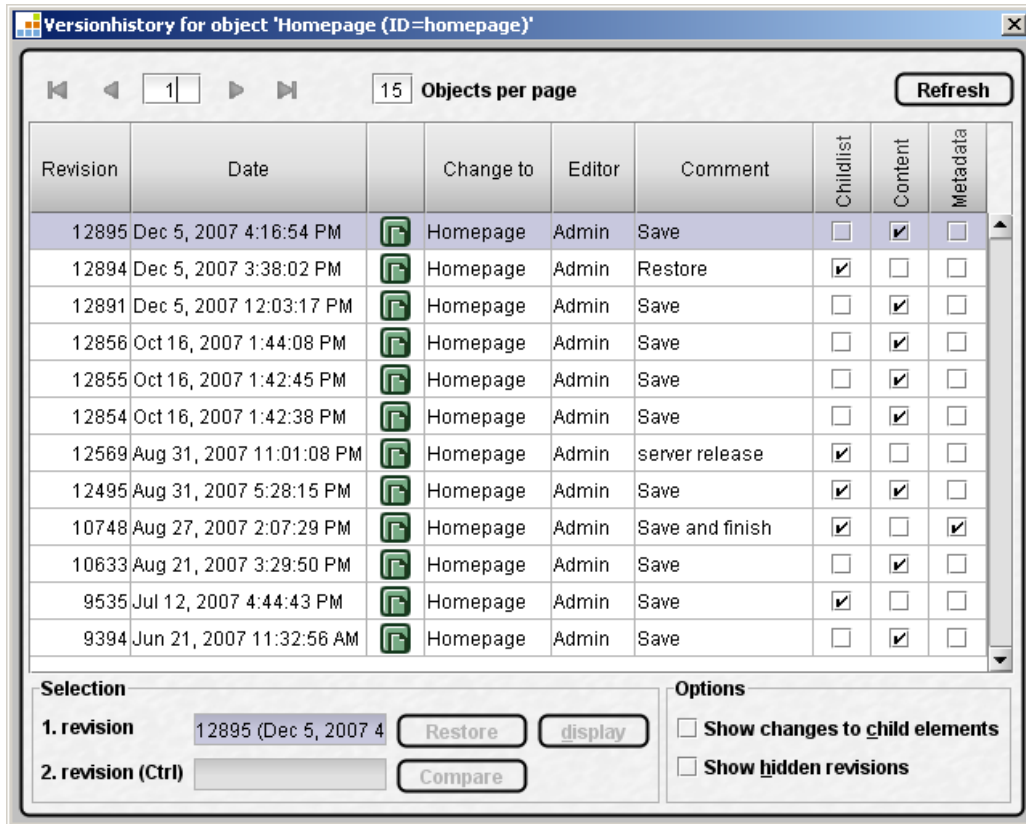
11.12.1.3 Supported objects

The version history is available for the following stores and objects:

11.12.2 Version history functions

Figure 11-97: Version history in the Page Store

The version history is divided into the following areas:

> ! *For information on the special features of the version history functions in conjunction with the archiving function, see also the FirstSpirit Manual for Administrators, "Project Archiving" chapter.*

11.12.2.1 Navigation area
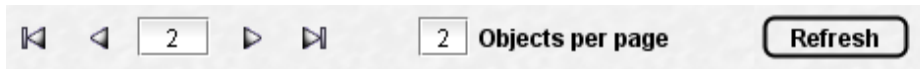
Figure 11-98: Version history – Navigation

The version history view can be adjusted in the Navigation area. The user is also shown the page on which they are currently located.

The list of the existing revisions can be paged through using the buttons in the left-hand area. The buttons are only active if more than one page of revisions exists:

◁▷ Go to first (most up-to-date) or last page of the version history.

◁▷ Go one page forwards or one page back.

The numbering of the page currently shown in the version history is displayed in the input field. The input field can also be used to directly enter page numbering. When the input is confirmed with Return, the focus then switches directly to the required page.

The **Objects per page** input field (see Figure 11-98) can be used to change the number of revisions displayed on one page of the version history. The view is updated when the entries are confirmed (with Return).

Refresh The version history is a static view. If the revisions of an object change (e.g. by another user saving a change), the view that has been opened is not automatically updated. Click this button to synchronise the revision history of the FirstSpirit server with the open revision view.

11.12.2.2 Display of an Object's Revisions

Revision	Date		Change to	Editor	Comment	Childlist	Content	Metadata
12895	Dec 5, 2007 4:16:54 PM		Homepage	Admin	Save	☐	☑	☐
12894	Dec 5, 2007 3:38:02 PM		Homepage	Admin	Restore	☑	☐	☐

Figure 11-99: Version history table view (Page Store example)

The table view shows all the revisions of an object, starting from the most up-to-date revision through to the oldest revision.

Revision: Revision number of the displayed object (see Chapter 11.12.1.2 page 393)

Date: Date of the last changes to the object, i.e. the time at which a new revision number was issued for the object.

Symbol: Icon with which the object is also displayed in the project's tree view.

Change to: Name with which the object is also displayed in the project's tree view. The displayed name depends on the "Extras / Tree View" setting, or from FirstSpirit Version 4.2 the "View / Preferred display language" setting (see Chapter 3.1.4.2 page 69).

Editor: Name under which the editor was authenticated on the FirstSpirit server.

Comment: Automatically issued comment which describes the change through which a new revision of the object was created (e.g. by saving the page).

> ⚠️ *Comments assigned by the modifier displayed (see also Figure 3-44). If comments were entered when workflows were started or forwarded, these are also displayed in this column (exception: in the Content Store).*

Further information is displayed depending on the store in which the version history is called and on which object it is called (see the relevant sub-chapters):

- In the Page Store on pages   (see Chapter 11.12.3 page 405)
- In the Content Store on data sets   (see Chapter 11.12.4 page 407)
- In the Media Store on media   (see Chapter 11.12.5 page 409)
- In the Site Store on page references   (see Chapter 11.12.6 page 411)
- In the Template Store   (see Chapter 11.12.7 page 413)

11.12.2.3 Restoring revisions

In the bottom part of the page it is possible to restore a revision selected in the table view.
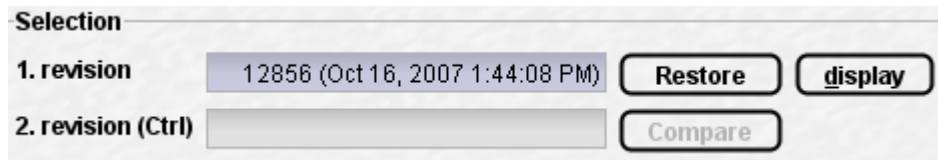
Figure 11-100: Version history – Restoring revisions

Restore If an older revision of an object has been highlighted in the table overview, click this button to restore the selected revision of the object. The options for a restore action can be selected in the following pop-up window:
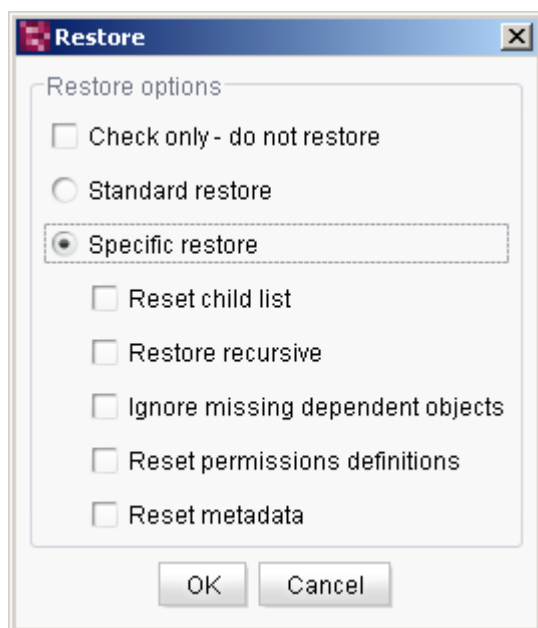
Figure 11-101: Restore – Options

Check only – do not restore: If this option is selected, the system checks whether the object can be restored without errors. To do this, the restore action is simulated, but the revision is not restored. A pop-up window then appears showing whether the object can be restored or not.

Standard restore: This option is set as a default. If the restore action is performed with this option, the selected revision is restored directly depending on the object.

Therefore, different options can be selected in the "Specific restore" area, depending on the object.

Specific restore: This option can be selected to manually adjust the standard restore options.

Specific restore – Reset child list: If this option is selected, the child list of the selected object is also reset to the status of the selected revision.

Specific restore – Restore recursive: If this option is selected, all the selected options are applied on all objects (children) located below the selected object.

Specific restore – Ignore missing dependent objects: If this option is selected, the missing references to the selected object are ignored when the object is restored.

> *This option is available to project administrators only.*

> *If the project has already been archived, this option is preset as a default and cannot be disabled if the revision to be restored lies within the archived period.*

Specific restore – Reset permissions definitions: If this option is selected, the permissions on restoring are reset to the status of the revision concerned. If this option is not selected, the permissions currently valid for the object are retained.

Specific restore – Reset metadata: If this option is selected, the object's metadata is reset to the status of the revision concerned. If this option is not selected, the current metadata of the object are retained.

While the restore action is being performed, a detail window appears which displays the progress of the actions. After performing the actions, further information can be read here.

If the restore is successful, the "The restore was successfully performed." message is closed by clicking the **View details** button. Click the **OK** button to close the detail window also. Following a failed restore action, the "Restore failed" message appears. This can be closed by clicking "OK".

The restored version is then displayed as the new most up-to-date version in the table and is labelled with the comment "Restored". The "restore" function is disabled on the respective most up-to-date revision.

Restore If an older revision of an object has been highlighted in the table overview, click this button to restore the selected revision of the object.

This restored version is then displayed as the new most up to date version in the table and is labelled with the comment "Restored". The "Restore" function is deactivated in the respective current revision.

Revision	Date		Change to	Editor	Comment
9398	**Jul 10, 2007 11:54:20 AM**		**Bildtest**	**Admin**	**revert**
8662	Apr 3, 2007 5:19:02 PM		Bildtest	Admin ...	Save
8658	Apr 3, 2007 5:18:07 PM		Bildtest	Admin ...	Create

Figure 11-102: Restore an older revision

> **!** *The following applies to all objects displayed or restored via the version history: The object itself can be displayed or restored with the older or newer revision; however, the content of the object can differ from the selected status. For example, if a content projection (from the Content Store) is restored, the current content of the data sets (both the number and the editorial content) do not correspond to the content of the data sets at the time of the revision. This behaviour applies to all dependencies within an object, i.e. e.g. also page references and the dependent pages. The relevant content should therefore be explicitly checked after being restored.*

11.12.2.4 Display older revisions (historic preview)

Selection

1. revision	12856 (Oct 16, 2007 1:44:08 PM)	**Restore**	**display**
2. revision (Ctrl)			Compare

Figure 11-103: Version history – Restoration of revisions

display If an object is selected in the table view click on this button to display a preview of the selected revision of this object.

11.12.2.5 Version Comparison - Selection of revisions

Changes to an object which have taken place from one revision number to another revision number can be displayed using difference visualisation. To this end two revisions in the history to be compared with each other are selected within the Version History dialog.

In the Selection area two revisions can be compared by simply clicking on the required revision and clicking on another revision with the CTRL key pressed (in the table).

It is not necessarily the last two revisions that are compared with each other, any two revision numbers in the table can be selected.

Both revisions are displayed in the bottom area of the dialog in the fields **1. revision** and **2. revision**.

Figure 11-104: Version history – Select revisions for comparison

[Compare] When this button is clicked the versions are compared and the dialog "Version Comparison" will open.
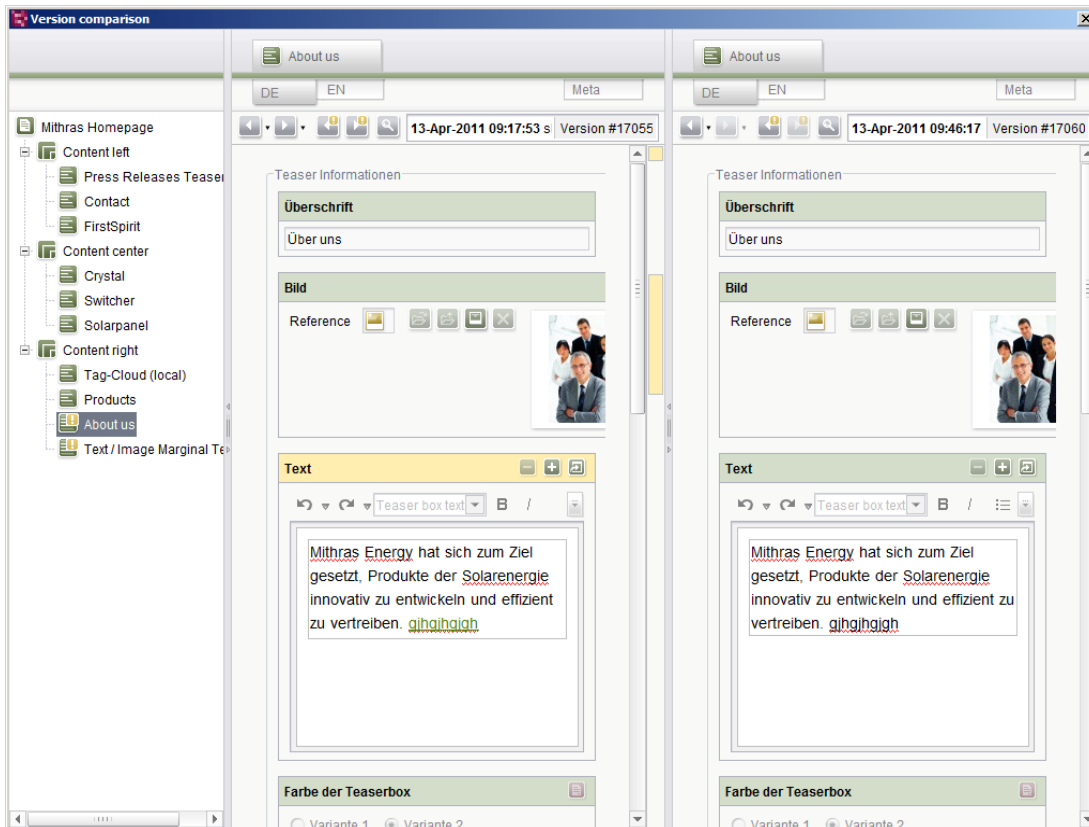
Abbildung 11-105: Dialog version comparsion (Page-Store)

The dialog is divided into three columns. In the left-hand column it is possible to further specify the relevant project elements for the version comparison. The tree structure can be used e.g. to select a single section of the page for the comparison.

The respective selected revisions are displayed in the two columns on the right. The revision information such as version number, date, time, last modifier and comment entered for the revision, are displayed at the top in the respective column. The Next and Back icons can be used to switch to the previous or next revision ("Revision selection") or the previous or next change ("Display changes") to the respective object.

The changes for the respective selected areas are displayed in the bottom area of the both columns on the right-hand side. Input components whose contents have changed compared to other versions of the object are highlighted in yellow. If a new section has been created in the content area of a page, an empty column for comparison will be shown for an older comparison version.
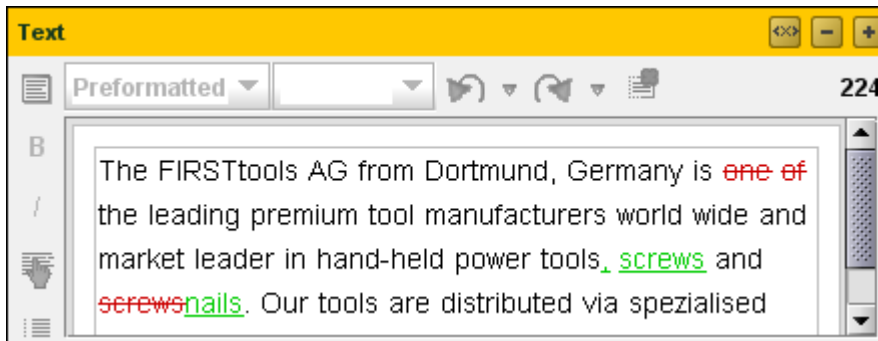
Figure 11-106: Changes (Version comparison)

The changes within the input component DOM-Editor will be marked separately, deleted (highlighted red) as well as added content is displayed (highlighted green).

Changes in the input component "Liste new (FS_LIST)" (see Chapter 10.26 page 307) are visualised separately, too. New entries are shown in green, modified in yellow and deleted in red colour. Entries are also marked as being changed (yellow) if the sort order of the entries has been changed for example.

> **!** *This dialog serves only to compare two revisions. A revision may not be subsequently edited or changed. It can therefore compare the version of any content from one version to another version will be adopted or modified.*

> **!** *In V4.2R4 and higher, a ribbon between the window areas shows in which input component changes exist. Yellow marking indicates that data has been changed, red means that data has been removed and green that new values have been added. A tooltip displays the name of the input component concerned and it is possible to switch directly to it with a click.*

> ❗ *The default input component FS_BUTTON is always hidden in a version comparison.*

11.12.2.6 Selection area for the comparison
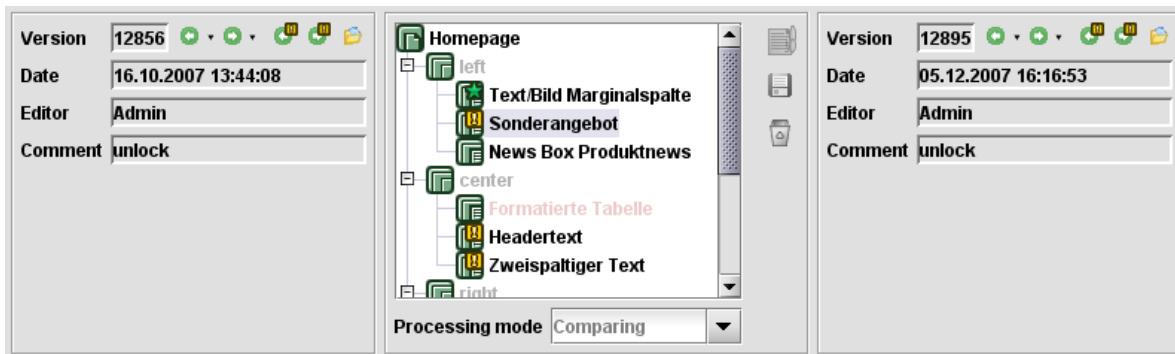
Figure 11-107: Selection area (version comparison)

The selection of a project element for version comparison can be further specified in this area. To this end, e.g. individual sections of the page can be selected and subjected to a comparison. This is especially necessary if the editorial contents are not entered at page level but at section level instead.

11.12.2.7 Display area for the change

The changes for the respective selected areas are displayed in this area. Input components whose contents have changed compared to other versions of the object are highlighted in yellow.

Figure 11-108: Changes (version comparison)

The changes within the DOM input component are specially highlighted, whereby

both deleted (marked in red) and added contents (marked in green) are displayed.

> ⚠️ *This dialog is only used to compare two revisions. A revision cannot be subsequently edited or changed. Therefore, in the version comparison it is not possible to copy contents from one version to another version or to change the contents.*

11.12.2.8 Options

Depending on the store and the object on which the version history is opene, further options of the version history are shown in the down right area of the window (see respective sub-chapters):

- in the Page Store on pages   (see Chapter 11.12.3 page 405)
- on the Content Store on data records   (see Chapter 11.12.4 page 407)
- in the Media Store on media   (see Chapter 11.12.5 page 409)
- in the Site Store on page references   (see Chapter 11.12.6 page 411)
- in the Template Store   (see Chapter 11.12.7 page 413)

11.12.3 Version history in the Page Store

11.12.3.1 At page level

Within the Page Store version management is available at page level. It is opened via the context menu of the required page (see Chapter 4.2.9 page 121). Click the "Version History" entry to open the dialog:

Figure 11-109: Version history at page level

The view of the version history can be adjusted in the top part of the window (see Chapter 11.12.2.1 page 396).

The table lists the revisions of an object and, apart from the generally available information (see Chapter 11.12.2.2 page 396), the following information is also displayed for the pages of the Page Store:

Child list: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the page's child list (e.g. deleting or adding a section).

Content: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the page's content (e.g. editing a section).

Metadata: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the page's metadata (e.g. definition of access rights via the metadata).

Selection:
For information on restoring a revision, see Chapter 11.12.2.3 page 398. For details of comparing two revisions of a page, see Chapter 11.12.2.4 page 400).

<u>Options:</u> Other changes concerning the current object can be selected for display in the Options area.

Show changes to child elements: If this checkbox is activated, in addition to each revision, the changes which took place on the page's child list within this revision are displayed. These changes are displayed in the table in grey.

12894	Dec 5, 2007 3:38:02 PM		Homepage	Admin	Restore	✔	☐	☐
12894	Dec 5, 2007 3:38:02 PM		right	Admin	Restore	☑	☐	☐
12894	Dec 5, 2007 3:38:02 PM		center	Admin	Restore	☑	☐	☐
12894	Dec 5, 2007 3:38:02 PM		left	Admin	Restore	☑	☐	☐
12894	Dec 5, 2007 3:38:02 PM		News Box P…	Admin	Restore	☑	☐	☐
12894	Dec 5, 2007 3:38:02 PM		Abstandhalter	Admin	Restore	☑	☐	☐
12894	Dec 5, 2007 3:38:02 PM		Gewinnspiel	Admin	Restore	☑	☐	☐

Figure 11-110: Version history – Show changes to child elements

Show hidden revisions: If this checkbox is activated, additional internal system revisions of an object are displayed (if they exist). If an archiving job has been performed (see *FirstSpirit Manual for Administrators*), this checkbox can be used to show elements of the version history whose content has been partly archived. If hidden revisions of the object only exist, these are displayed directly on opening the version history. Hidden revisions cannot be restored.

Show partly archived revisions: If an archiving job has been completed (see *FirstSpirit Manual for Administrators*), this option can be activated to show all revisions of objects which are still completely retained but whose revision number is smaller than that of the smallest/last revision not yet archived. This can be the case, for example, if pages from the Page Store are archived in an archiving job, but not the page templates on which they are based. Partially archived revisions are shown in the table in grey. The relevant buttons can be used to perform the "Restore", "Display" and "Compare" functions on them.

11.12.4 Version history in the Content Store

11.12.4.1 At data set level

Within the Content Store the version history is available on data sets. The version history is called using the context menu of the required data set (see Chapter 5.2.2 page 140) or click the ⬚ button in the data set's edit window.

Figure 11-111: Version history at data set level

The view of the version history can be adjusted in the top part of the window (see Chapter 11.12.2.1 page 396).

The table lists the revisions of the data set and at the same time, apart from the generally available information (see Chapter 11.12.2.2 page 396) for a data set, other information is also displayed (e.g. category).

Selection:

For information on the restoring a revision, see Chapter 11.12.2.3 page 398. For detailed of comparing of two revisions of a data set, see Chapter 11.12.2.5 page 401.

From FirstSpirit Version 4.1, the following window is displayed with the following information:
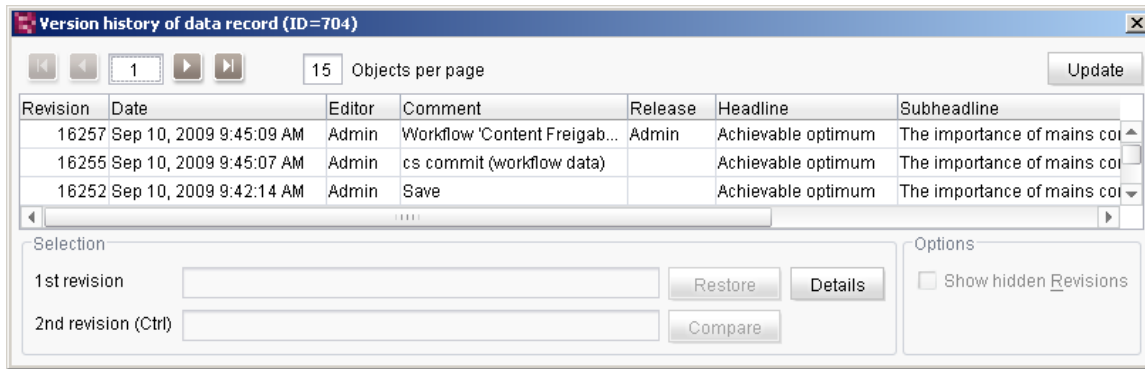
Figure 11-112: Version history at data set level

In addition to the generally available information (see Chapter 11.12.2.2 page 396), the respective revision of all data entered for a data set is listed here in tabular form. If an archiving job has been performed (see *FirstSpirit Manual for Administrators*), the "Show hidden Revisions" checkbox can be used to show elements of the version history whose content has been partly archived. If hidden revisions of the object only exist, these are displayed directly on opening the version history.

> ❗ *New data sets created which have not yet been saved do not yet have a version history. The context menu and the button are therefore hidden.*

11.12.5 Version history in the Media Store

11.12.5.1 At media level

Within the Media Store the version management is available at media level. It is opened via the context menu of the required medium (see Chapter 6.2.9 page 167). Click the "Version History" entry to open the following dialog:
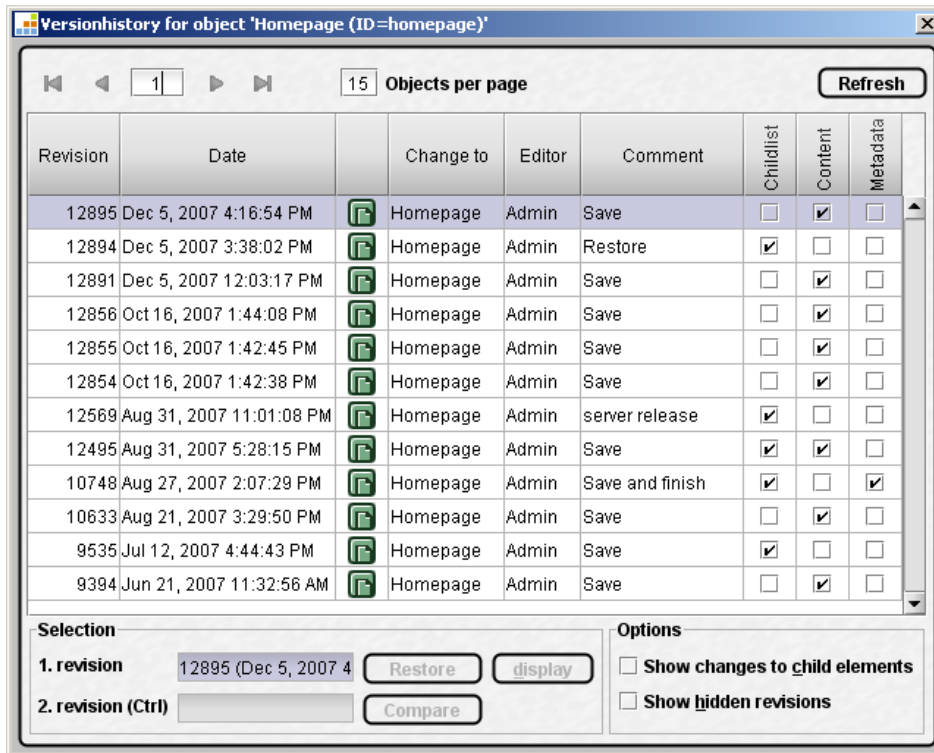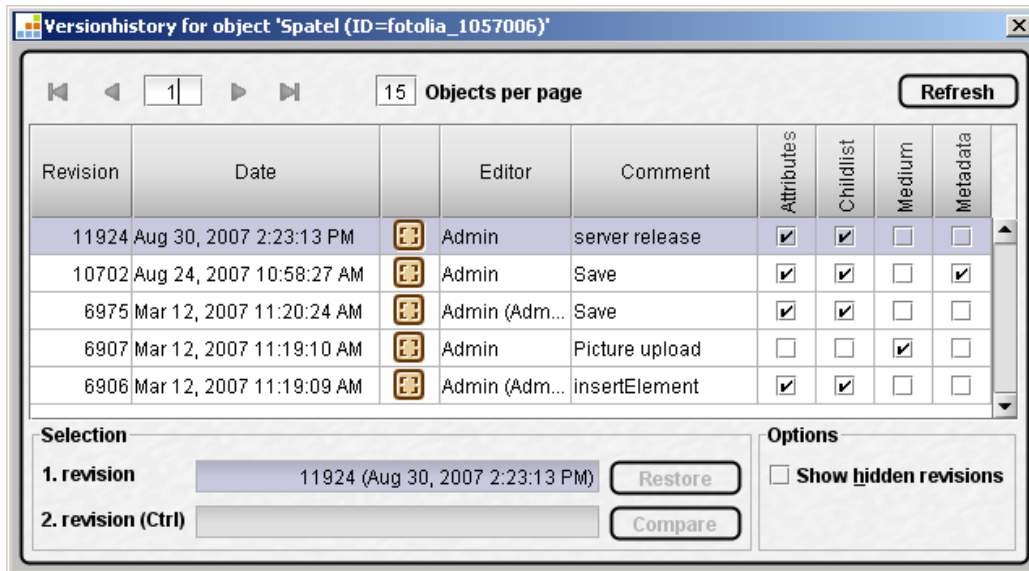
Figure 11-113: Version history at media level

The view of the version history can be adjusted in the top part of the window (see Chapter 11.12.2.1 page 396).

The table lists the revisions of the medium and apart from the generally available information (see Chapter 11.12.2.2 page 396) for a medium, other information is also displayed:

Attributes: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the medium's attributes (e.g. when a medium is released the release status changes).

Child list: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the medium's child list (e.g. rendering a new resolution).

Medium: If this checkbox is activated, the change which led to assignment of a new revision number involves a direct change to the medium's content (e.g. uploading a new picture file).

Metadata: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the medium's metadata (e.g. definition of access rights via the metadata).

Selection:
For information on restoring a revision, see Chapter 11.12.2.3 page 398. For details of comparing two revisions of a page, see Chapter 11.12.2.5 page 401).

Options: Other changes concerning the current object can be selected for display in the Options area.

Show hidden revisions: If this checkbox is activated, additional internal system revisions of an object are displayed (if they exist). If an archiving job has been performed (see *FirstSpirit Manual for Administrators*), this checkbox can be used to show elements of the version history whose content has been partly archived. If hidden revisions of the object only exist, these are displayed directly on opening the version history. Hidden revisions cannot be restored.

Show partly archived revisions: If this option is activated, all revisions of objects are shown, which are still completely retained, but whose revision number is smaller than that of the smallest/last not yet archived revision. Partially archived revisions are shown in the table in grey.

11.12.6 Version history in the Site Store

11.12.6.1 At page reference level

The version management is available in the Site Store at the level of a page reference. It is opened via the context menu of the required page reference (see Chapter 7.2.9 page 221). Click the "Version History" entry to open the following dialog:
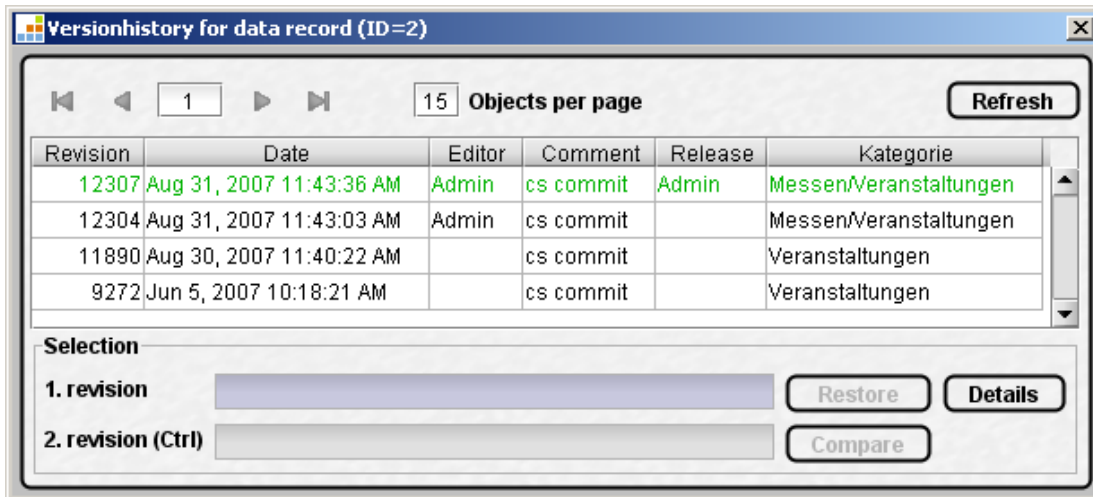

Figure 11-114: Version history at page reference level

The view of the version history can be adjusted in the top part of the window (see

Chapter 11.12.2.1 page 396).

The table lists the revisions of the object and, apart from the generally available information (see Chapter 11.12.2.2 page 396) for a page reference, other information is also displayed:

Attributes: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the page reference's attributes (e.g. the release status changes with a release).

Child list: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the page reference's child list.

Metadata: If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the page reference's metadata (e.g. definition of access rights via the metadata).

Selection:
For information on restoring a revision, see Chapter 11.12.2.3 page 398. For details of comparing two revisions of a page reference, see Chapter 11.12.2.5 page 401.

Options: Other changes concerning the current object can be selected for display in the Options area.

Show changes to pages: If this checkbox is activated, changes to the page to which the page reference refers are also displayed.

Show hidden revisions: If this checkbox is activated, additional internal system revisions of an object are displayed (if they exist). If an archiving job has been performed (see *FirstSpirit Manual for Administrators*), this checkbox can be used to show elements of the version history whose content has been partly archived. If hidden revisions of the object only exist, these are displayed directly on opening the version history. Hidden revisions cannot be restored.

Show partly archived revisions: If this option is activated, all revisions of objects are shown, which are still completely retained, but whose revision number is smaller than that of the smallest/last not yet archived revision. Partially archived revisions are shown in the table in grey.

> ❗ *The following applies to all objects displayed or restored via the version history: The object itself can be displayed or restored with the older or newer revision; however, the content of the object can differ from the selected status. For example, if a content projection (from the Content Store) is restored, the current content of the data sets (both the number and the editorial content) do not correspond to the content of the data sets at the time of the revision. This behaviour applies to all dependencies within an object, i.e. e.g. also page references and the dependent pages. The relevant content should therefore be explicitly checked after being restored.*

11.12.7 Version history in the Template Store

Within the Template Store the version management is available for all elements (templates, scripts, etc.). For details of version history in the Template Store, see the *FirstSpirit Manual for Developers*.

11.13 Show dependencies via reference graphs (from V4.1)

> ❗ *This function is available for FirstSpirit Version 4.1 and higher only. Screenshots are therefore displayed in the new "LightGray" look & feel. The display can differ slightly in the "Classic" look & feel.*

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 further developed. The reference graph of a project is used to recognise dependencies within the project and is therefore an essential component of complex functions, for example, the server-side release.

Reference graphs can be requested via the **Extras / Display Dependencies** context menu of an object (see Figure 11-115). Reference graphs of individual data sets of the Content Store are called via the context menu of the respective data set.

> ❗ *This function is available to project administrators only.*

The tab in which opening windows display the dependencies of the object in the form of incoming and outgoing edges, both for the current status (Current Status tab) and

for the last released status (Release Status tab), provided the project uses the release option:
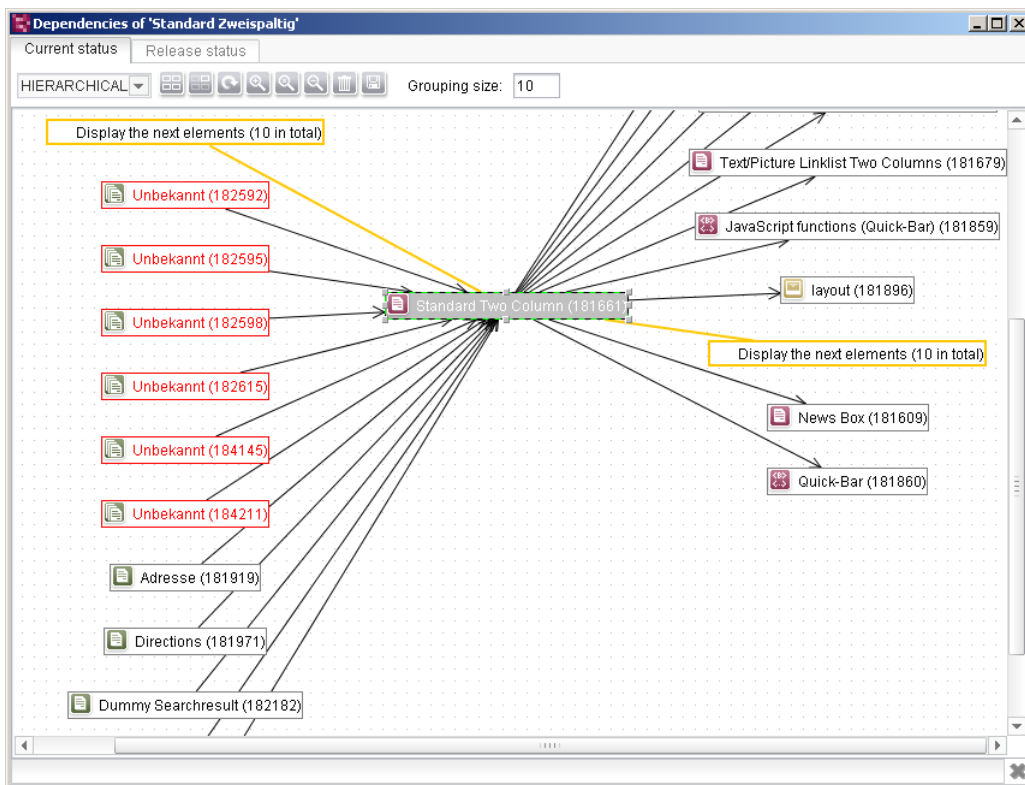
Figure 11-115: Displaying dependencies via the reference graph

Each object to which a dependency exists is displayed with ID and corresponding object icon. Invalid references are marked by a red border and red lettering. Double-click "Show the next elements" to show other dependent elements.

Arrows indicate whether the references are incoming or outgoing. Double-click an element to show the references to this object as well. Individual elements can be selected by right-clicking, to select several elements, the <Ctrl> and <Shift> key must be pressed simultaneously.

HIERARCHICAL: The hierarchical view displays the incoming and outgoing references grouped in a kind of tree structure. It is especially advisable for complex dependencies (see Figure 11-115).

ORGANIC: The organic display shows the dependent elements around the source object. Loading this view is more time-intensive than loading the hierarchical view and it should therefore only be used for objects with few dependencies.

Layout: this icon is used to apply the layout to all references.

Layout (selection): this icon is used to apply the layout to selected references only.

Update: if the object's dependencies change while the reference graph is open, the changes can be displayed using this icon.

Increase zoom: this icon can be used to increase the zoom, to enable a section of the reference graph to be examined in greater detail.

Zoom 1:1: this icon can be used to reset the view of the reference graph to its initial status.

Reduce zoom size: this icon can be used to reduce the zoom so that a larger area of the reference graph can be examined.

Hide: this icon can be used to hide the reference graph or parts of it. To hide individual dependencies, the elements which are to be hidden are selected with the click of a mouse beforehand.

Save as picture: this icon can be used to save the view on the workstation computer in the .png format for use at a later date.
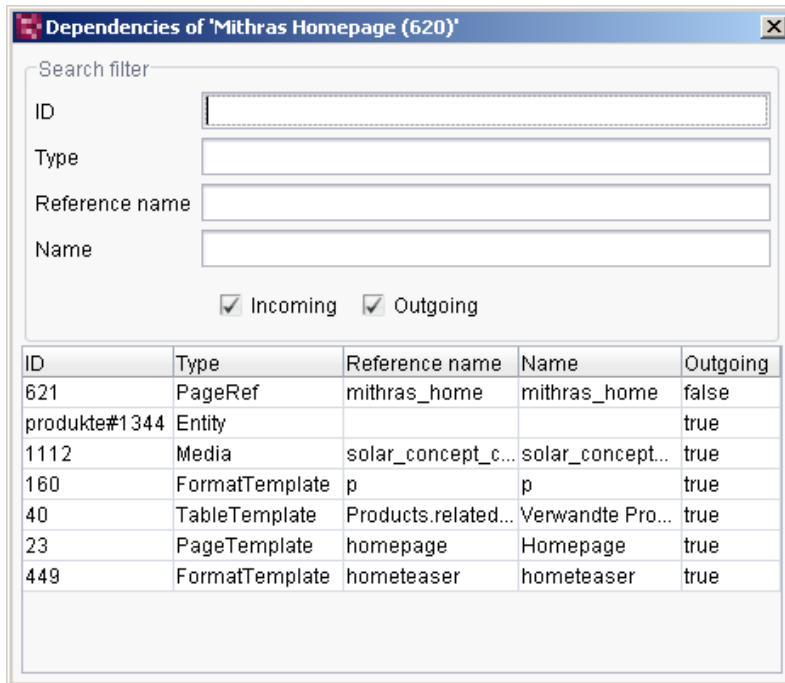
Grouping size: As a default, when the reference graph is opened, only the first 10 dependent elements are displayed as a maximum. This input field can be used to change the maximum number. A re-defined grouping size does not have an effect until the next time the "Display the next elements" function is activated.

The context menu of an object (right-click) can also be used to perform the following functions:

Copy ID: This function is used to store the ID of the respective object in the temporary memory.

Display list: This function is used to display the dependent objects in list form:

Figure 11-116: Reference graph – Display list

The dependent elements can be filtered in this list by ID, object type, reference name and name. The checkboxes can be used to hide incoming or outgoing references. Click the heading fields to display the entries sorted by the values in the corresponding column.

Jump to element: Use this function to switch directly to the object in JavaClient.

> ⚠ *From FirstSpirit Version 4.2, format templates used within the CMS_INPUT_DOM and CMS_INPUT_DOMTABLE input components can also be displayed via the reference graphs.*

11.14 Remote access from one project to another project

FirstSpirit supports remote access to other FirstSpirit projects. Remote access is only possible if the relevant remote project(s) have been configured by the project administrator in the target project.

There are different types of remote access:

- Type: "Related Projects" (see Chapter 11.14.1 page 417).

▪ Type: "Remote Media" (see Chapter 11.14.3 page 418).

In most cases remote access takes place via specific, specially configured input components.

11.14.1 "Related Projects" concept

The "Related Projects" concept handles links from one project in another FirstSpirit project. These links can be realised indirectly using the definition of link targets in a FirstSpirit project (via specially configured input components), as well as by means of direct definition within the Site Store.

If a valid licence is available for the function the "Related Projects" access can be activated via the FirstSpirit Server and Project Configuration.

> **!** *The "Related Projects" function is a licence-dependent additional module.*

11.14.2 Use options for "Related Projects"

11.14.2.1 Create a link to a related project in the structure

Unlike all other remote access options in this case it is not necessary to use any specially configured input components. The URL in the required remote project can be defined directly at page reference level using the "Links" setting. To this end the "Related Project" must be activated first. The required page reference of the remote project can then be selected (see Chapter 7.6.1.3 page 237).

11.14.2.2 Select link target to a related project

Link targets (Site Store and/or Media Store) from the remote project are selected using a specially configured link input component (see Chapter 10.23.4). Depending on the configuration, this input component can be used to reference page references or media from the local store and/or the store of the remote project (for the "sitestoreref" attribute only).

If appropriately configured by the template developer, selection of remote link targets is supported for the link types: "internal link" (see Chapter 10.23.1) and "external link" (see Chapter 10.23.2).

Define link target ("sitestoreref" attribute): The "Related Projects" function can be used here to select not only page references and media from the local Store and/or Media Stores but also page references and media from a remote project (see Chapter 10.23.4).

Select Picture ("mediaref" attribute): Apart from the option of selecting a link target ("sitestoreref" attribute) it is also possible to select a media object ("mediaref" attribute), which is linked e.g. instead of a text (cf. "Link Text" field in Figure 10-53). The "Related Projects" function does not have any effect on the selection of a picture link. Remote access for this selection is only available if the "Remote Media" function has been configured (cf. Chapter 11.14.1).

Selection of a link target to a related project is supported for the following input components:

- CMS_INPUT_DOM   (see Chapter 10.11 page 276).
- CMS_INPUT_LINK   (see Chapter 10.23 page 291).
- CMS_INPUT_LINKLIST (see Chapter 10.24 page 304).

11.14.3 "Remote Media" concept

The FirstSpirit "Remote Media" function is an add-on to the licence-dependent "Package Management" function. "Remote Media" is an additional option for projects with an extensive Media Store.

The aim of the "Remote Media" concept is to create all media in a separate media project and to centrally manage them there. All FirstSpirit projects involved can then access this project's media data (pictures and files) via the remote media access.

Unlike distribution of the media via package management the media do not have to be imported into the projects involved, but instead can be directly referenced via the

remote media access. The objects remain physically in the media project but can be used in all the required projects (for further information on package management, see "FirstSpirit PackagePool" documentation).

The advantages of the "Remote Media" function are:

- No additional storage requirement for media to be used in several projects.

- Simplified updating and management as all media are located in a central media project.

- Shorter generation time for the target projects involved.

If a valid licence is available for the function the remote media access can be activated via the FirstSpirit Server and Project Configuration.

> **!** *The "Remote Media" function is a licence-dependent additional module.*

11.14.4 Use options for "Remote Media"

11.14.4.1 Picture input component (CMS_INPUT_PICTURE)

Picture files are selected from the remote project using a specially configured picture input component. Depending on the configuration and the access permissions, this input component can be used to reference picture files from the local Media Store and/or the Media Store of the remote project (see Chapter 10.10.1).

In addition, picture files from the local file system can be uploaded into the Media Store of the remote project in specially configured input components (see Chapter 10.10.2).

> **!** *If the picture was selected via the remote project's Media Store it only exists in the target project via this reference, i.e. it cannot be found in the target store's Media Store.*

11.14.4.2 File input component (CMS_INPUT_FILE)

Files are selected from the remote project using a specially configured file input component. Depending on the configuration and the access permissions, this input component can be used to reference files from the local Media Store and/or the Media Store of the remote project (see Chapter 10.10.1).

In addition, files from the local file system can be uploaded into the Media Store of the remote project in specially configured input components (see Chapter 10.10.2).

> **!** *If the medium was selected via the remote project's Media Store it only exists in the target project via this reference, i.e. it cannot be found in the target project's Media Store.*

11.14.4.3 Select reference graphic from a remote project

References to media objects from the remote project are selected using a specially configured link input component (see Chapter 10.23.4). Depending on the configuration, this input component can be used to reference media from the local Media Store and/or the Media Store of the remote project (for the "mediaref" attribute only).

If appropriately configured by the template developer, selection of remote picture links is supported for the link types: "internal link" (see Chapter 10.23.1), "external link" (see Chapter 10.23.2) and "content link" (see Chapter 10.23.3).

Define link target ("sitestoreref" attribute): The "Remote Media" function does not have any effect on the selection of a link target. Remote access for this selection is only available if the "Related Projects" function has been configured (cf. Chapter 11.14.1).

Select Picture ("mediaref" attribute): Apart from the option of selecting a link target ("sitestoreref" attribute) it is also possible to select a media object ("mediaref" attribute), which is linked e.g. instead of a text (cf. "Link Text" field in Figure 10-53). Apart from selecting media from the local Media Store, the "Remote Media" function can also be used here to select media from a remote project (see Chapter 10.23.4).

Selection of a picture link from a remote project is supported for the following input components:

- CMS_INPUT_DOM   (see Chapter 10.11 page 276).

- CMS_INPUT_LINK   (see Chapter 10.23 page 291).
- CMS_INPUT_LINKLIST (see Chapter 10.24 page 304).

12 Workflows in the FirstSpirit JavaClient

A workflow is a sequence of tasks which are worked through in a fixed, specified structure. The tasks serve to convert an object, for example a page from the Page Store, from its initial state (e.g. "Page Changed") into a final state (e.g. "Changed Page Checked and Released"). Both due dates and authorised groups of people can be specified for th tasks to be executed between these two states.

The authorisation or permission to start a workflow is defined within the FirstSpirit editing environment using the "Permissions" dialog (see Chapter 13.2 page 462).

The structure (task sequence) and properties (e.g. context free) of a workflow and definition of the authorised persons or groups who may pass forward a workflow from one task to the following task are defined within the Template Store of the FirstSpirit JavaClient (for further information see FirstSpirit Manual for Developers (Part 1)) or by assigning special permissions for the individual steps of a workflow (see Chapter 13.2.5 page 469).

12.1 FirstSpirit Standard Workflows

In FirstSpirit there are two integrated workflows:

1. **Task** workflows for general completion of tasks within the project. Authorised persons or groups can then use this workflow to set a task and to assign the task to a specific person or group. They switch the workflow to the final "Finished" state when they have completed the task or are given the opportunity to ask questions about the task.

2. **Request Release** workflows for the release of new objects created or existing objects changed in the project. This workflow is described in detail in Chapter 12.3 page 425.

12.2   Starting a workflow

FirstSpirit recognises two types of workflows:

Context-Free workflows are not tied to a specific object within the project and are therefore started "without context", i.e. without reference to an object. The standard "Task" workflow is an example of a context-free workflow. (see Chapter 12.2.2 page 425).

Context-Bound Workflows are tied to a specific object within the project and are always started and executed with reference to this object (see Chapter 12.2.1 page 423). The standard "Request Release" workflow is an example of a context-bound workflow (see Chapter 12.2.1 page 425).

From FirstSpirit version 4.1 on workflows can be started on several objects at the same time. More than one object can be selected by highlighting the desired objects and clicking the SHIFT or CTRL jey simultaneously (see also FirstSpirit Manual for Developers (Basics)). Please bear in mind that there are only reduced options in the dialog for starting and switching the workflow (Figure 12-4 and Figure 12-5, see Chapter 12.3.1 page 426).

12.2.1   Context-Bound Workflows

A context-bound workflow can be opened in the context menu of a tree element using the general "Start Workflow" function (cf. Chapter 4.2.10 page 121).

Figure 12-1: Start Context-Bound Workflow

All context-bound workflows from the project's Template Store are listed under this

menu item.

> **!** *If the entries are inactive the person dealing with the task does not have the necessary permissions to start the workflow on the tree element (see Chapter 13.2 page 462).*

> **!** *From FirstSpirit version 4.1 on, according to the settings of the template developer workflows can be started via a user-defined keyboard shortcut.*

After a workflow has been started an action window appears. Various settings can be made or read in this action window (see Figure 12-4). The dialog box can be used to forward the selected object to the next step or state within the workflow, for example from the start state to a following state (for an example, see Chapter 12.3.1 page 426).

At each node within the tree structure, only one workflow can ever be active at any one time. It is therefore not possible to start a further workflow for an object without ending the already started workflow first or unless the workflow has reached the end state.

If a workflow has been started for an object the context menu entry changes. Instead of "Start Workflow" (cf. Figure 12-1) the menu now shows the "Workflow Action" entry:

Workflow Action | prüfen

Figure 12-2: Forward context-bound workflow

All actions which have to be executed to switch the active workflow to the next state are listed under this menu entry.

> **!** *If the entries are inactive the person dealing with the task does not have the necessary permissions to pass forward the workflow on the tree element (see Chapter 13.2.5 page 469).*

> ⚠️ *From FirstSpirit version 4.1 on, according to the settings of the template developer workflows can be started via a user-defined keyboard shortcut.*

After an action has been opened an action window appears again in which the various settings can be made or read.

Started schedules can also be passed forward via the task list (see Chapter 12.3.2 page 431).

12.2.2 Context-Free Workflows

A context-free workflow can be started using the "Start Workflow (without Context)" function in the "Tasks" menu of the FirstSpirit menu bar (see Chapter 3.1.2.2 page 54).
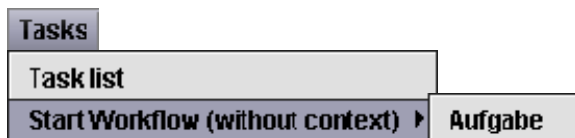
Figure 12-3: Start context-free workflows via the menu bar

Analogous to the execution of a context-bound workflow (see Chapter 12.2.1 page 423), a dialog box opens here too. The dialog box can be used to forward the selected object to the next step or state within the workflow, for example from the start state to a following state (for an example, see Chapter 12.3.1 page 426).

Each context-free workflow can be started any number of times within a project. If a context-free workflow has been started the workflow can be passed forward via the task list (see Chapter 11.1 page 314).

12.3 Release of objects (standard sequence)

In all FirstSpirit projects which carry out a release check objects, for example pages and media must be released after they have been created as new objects or have been changed. The release of objects in the FirstSpirit JavaClient is carried out by a workflow (cf. Introduction to Chapter 12 ff.). The "Request Release" workflow is intended to ensure that a new article or contribution created by the editor or a change to the existing contents is subjected to a check before the "Live

Transmission". Editors can use the standard FirstSpirit workflow "Request Release" which is described, step by step in the following. However, the release process described here can vary depending on which workflows are already established in the company or are to be established. A workflow can be started for all FirstSpirit objects. The user should have the permissions required to execute this workflow (see Chapter 13.1.4.9 page 460).

A release must be requested if:

- a new object has been created or
- an object has been changed

The object then has "changed" status and is indicated by red lettering in the tree structure in the left-hand side of the window. There are different options with which you can "Request Release":

- Request a release via the context menu (see Chapter 12.3.1)
- Request a release via the task list (see Chapter 12.3.2)

12.3.1 Request a release via the context menu

The context menu is opened at the object to be released (see Chapter 12.2.1 page 423). The editor opens the standard workflow using the "Start Workflow" context menu entry, sub-menu entry "Request Release".

The "Workflow Action" dialog window opens for starting the workflow:

Figure 12-4: Workflow Action (request release) dialog box

Various settings can be made or read in this action window.

User: The required editor for this workflow can be selected using the [icon] icon behind the row (for information on the assignment of users or groups, see Chapter 13.2.4 page 467).

Priority: This can be used to set the priority to be given to this workflow during the editing or processing.

Due date: A deadline by which the whole workflow must have been completed can be specified here.

Comment: This field can be used to enter a more detailed description of the task for the user. This comment will be shown to the next editor in the workflow dialog and adopted to the version history of the object.

Last Step: Information on the previously executed work step is displayed here.

Preview: Click the button to generate a preview of the object to be released in the browser. In this way changes can be simply checked before release and if necessary

can be revised.

Go to object: Click the button to change the focus in the JavaClient's edit window directly to the object to be released.

Actions Area: A button for each state which can be achieved is displayed in the "Actions" window area. The buttons displayed in this area change depending on the object's status. If the object has "changed" status the "Request" button is available here. If the object has the following "Release Requested" status the "Grant" and "Do not Grant" release buttons are available in this position.

Request: Click the button to request the release for the object. The lettering of the object in the tree structure in the left-hand part of the window now changes from red to blue.

Cancel: Click this button to close the dialog window; a release is not requested.

> [!] *If the workflow has been started on several objects only the options "Priority", "Due date", "Comment" and the actions are available.*

If a release has been requested the object is passed forward to the next status of the workflow. As a result, not only the colour of the lettering in the tree structure changes but the context menu too (see Chapter 12.2.1 page 423). The started "Request Release" workflow can be forwarded. Click the context menu "check" entry to open the "Workflow Action" dialog window again.

New buttons are now available in the "Actions" area of the window only.

Figure 12-5: Workflow action

Do not Grant: If the button is clicked the release is refused by the responsible editor. The object switches from "Release requested" status to "Object not released" status. The lettering of the object in the tree structure in the left-hand part of the window now changes from blue to red. The editor must now check their changes and if necessary revise them and then request a release again.

Grant: Click the button to grant the release for the object and exit the "Request Release" workflow. The lettering of the object in the tree structure in the left-hand part of the window now changes from red to black.

Cancel: Click this button to close the dialog window; the object's status is not changed.

> ! *If the workflow was forwarded for several objects simultaneously, only one comment can be assigned and only one action can be performed in this dialog.*

With FirstSpirit Version 4.2, the workflow dialog box was enhanced:
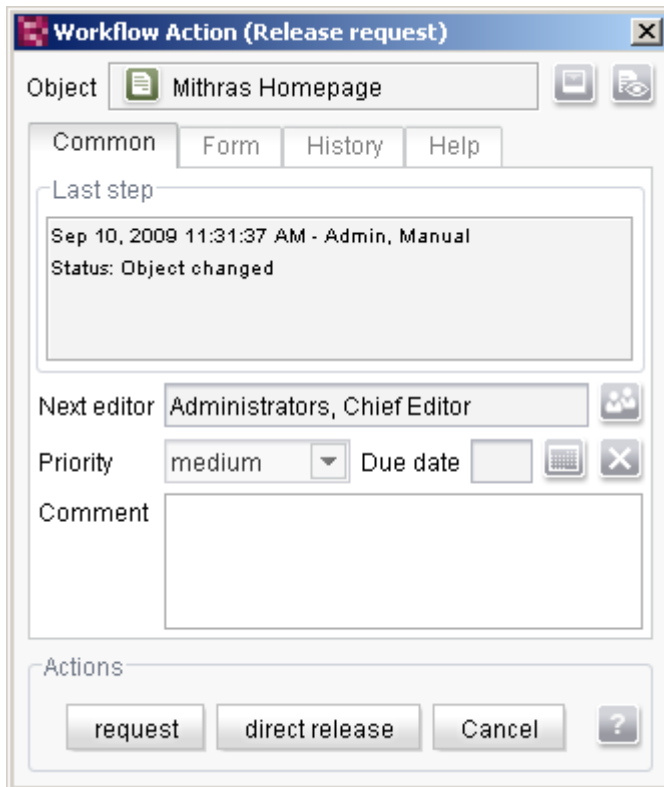
Figure 12-6: Workflow Action (request release) dialog box (from V4.2)

The following functions have been added:

Object: The object on which the workflow was called is displayed here again with the object symbol.

: jump to the object / display preview: these icons perform the function of the "Jump to object" and "Display preview" buttons

: compare: **From FirstSpirit Version 4.2R2** a comparison between different versions of the respective object can be opened using this icon (see also Chapters 11.12.2.5 from page 401). The **right-hand** side shows the current revision of the object, i.e. the current state of the object. The **left-hand** side shows the revision of the last release or (if the respective object has not been released yet) the last revision before the current revision.

"Help" tab: The editor can use the icon or tab within the action dialog to request further information on the action buttons 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):
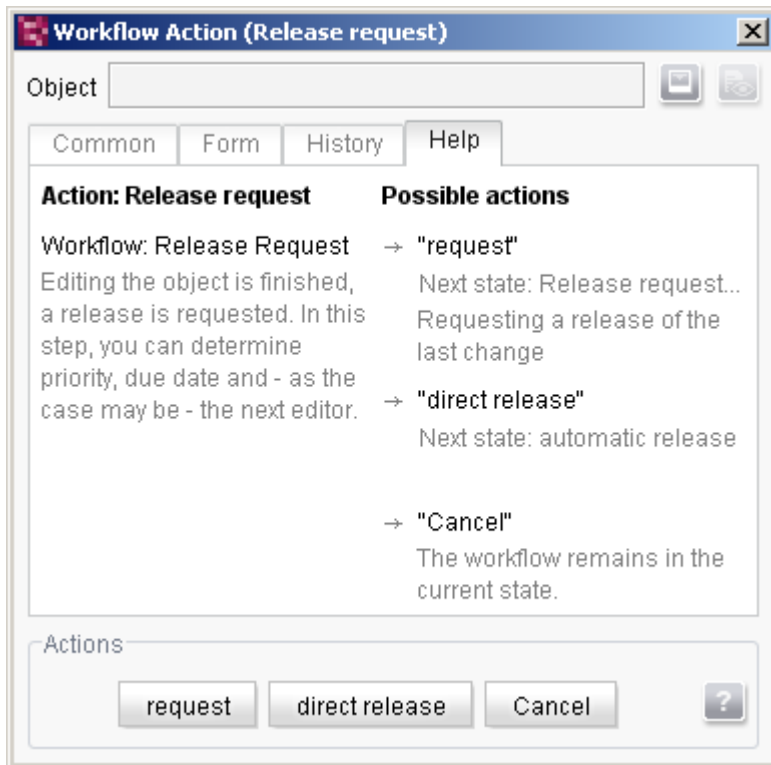
Figure 12-7: Workflow action – "Help" tab

The template developer can deposit the descriptive texts of the Help dialog within the workflow model using the language-dependent description. In addition, input fields, tooltips, elements of a combobox, etc. can be displayed language-dependent.

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

12.3.2 Request a release via the task list

To request a release via the task list the editor first opens the task list using the button in the toolbar or using the "Tasks / Task List" menu item of the menu bar.
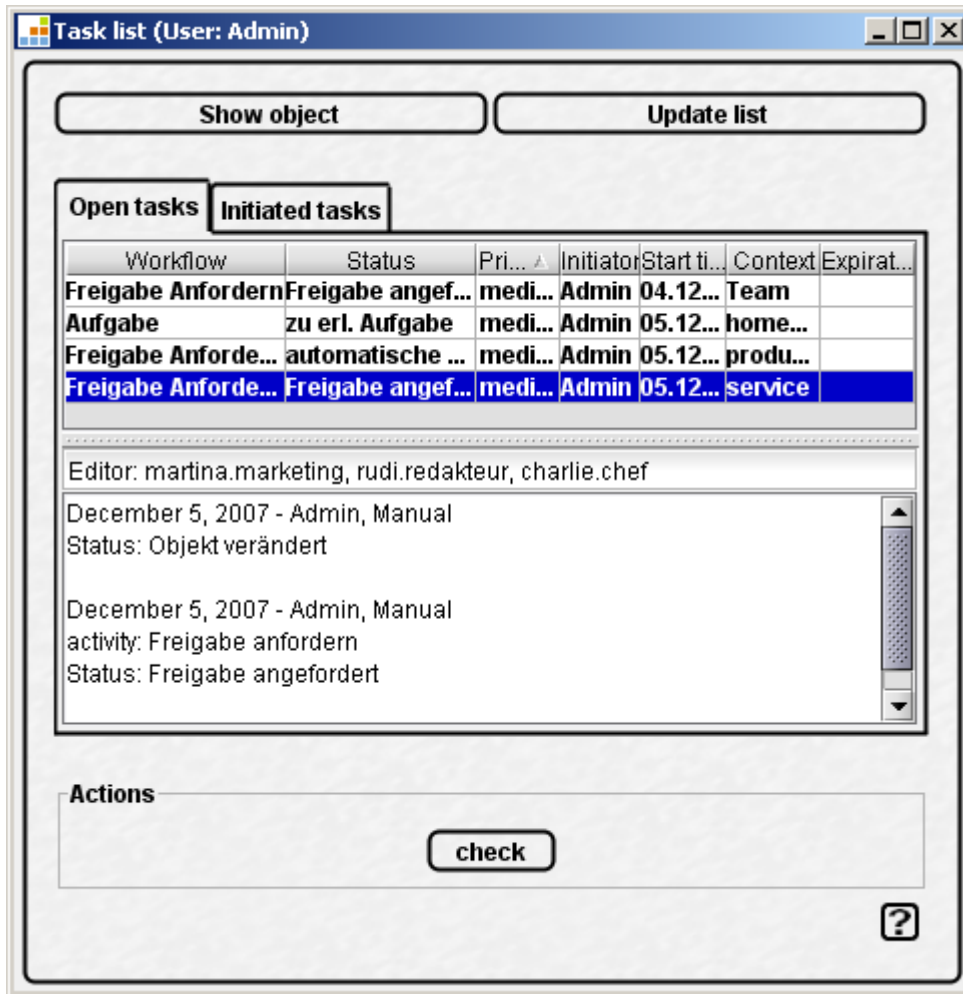
Figure 12-8: Task list

Open Tasks: All tasks assigned to the currently logged in user are displayed in the "Open Tasks" tab. It does not matter whether the task was assigned by the user themselves or by another user.

Started Tasks: All tasks assigned by the currently logged in user are displayed in the "Started Tasks" tab.

Show object: Click this button to change the focus in the JavaClient's edit window directly to the object highlighted in the list.

Update list: The list is not updated automatically. If several tasks have been dealt with in the meantime or new tasks have been added while the task list was open the list no longer shows the up to date status. Click this button to synchronise the task list of the FirstSpirit server with the open task list.

Actions: As soon as an object in the list is selected the buttons which can be executed for the selected object appear in the "Actions" part of the window. The

appropriate buttons for each object status are displayed in the "Actions" part of the window:

Status	possible actions (buttons)
Object changed	Request release
Release requested	Check (grant / do not grant)
Object not released	Edit
Object released	Final status

If a release has been requested for an object from the Open Tasks list the object is switched to the next workflow status. As a result, not only the colour of the lettering in the tree structure changes but the button in the "Actions" window area too. Click the "Check" button to open the "Workflow Action" dialog window again. The following release steps are analogous to the release of an object using the context menu (see Chapter 12.3.1 page 426).

12.4 Special release options

The special release options can be opened using the context menus of the various stores. Click the **Extras – Release** function at the required object to open a window with the various release options.
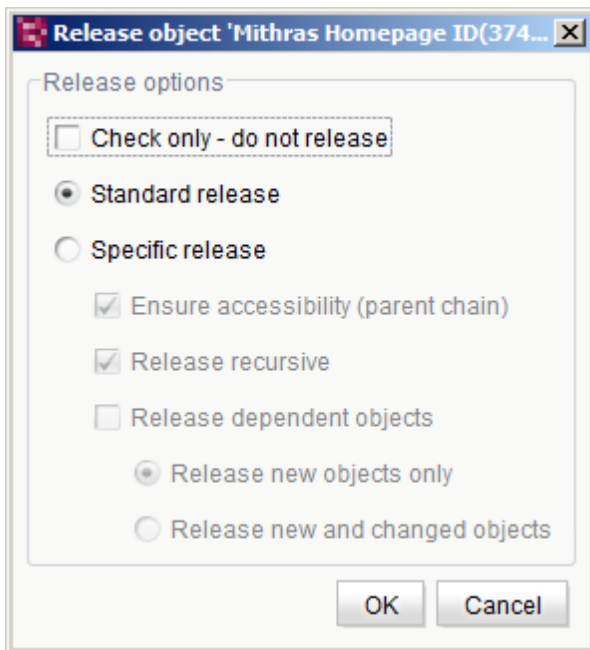

Figure 12-9: Release options for direct release

Comment: Similar to the assignment of comments when leaving the edit mode (see Chapter 3.2.6 page 92), this input field can be used to enter a comment when

releasing the object. This comment will be taken over to the version history, too.

> ❗ *If this input field is visible or not depends on the settings in the project configuration.*

Check only – do not release: If this option is selected the system checks whether a release can be carried out without errors. However, the release is not released.

Standard release: If this option is selected, then the current object (e.g. page or folder of the Page Store, picture from the Media Store) including additional, permanently defined release options, are directly released. The result of a "Standard release" depends on the element, i.e. a different release is performed on a folder than on a page.

Examples:

▪ The lower level sections and the parent elements which have never been released are also released on a page in the Page Store.

▪ In the case of a medium in the Media Store, the elements in the parent chain that have never been released yet are released.

▪ The standard release for a page reference in the Site Store only takes into account the page reference itself.

Under the greyed out specific release options it is possible to read which standard release options are currently active. However, the standard release options cannot be changed.

Specific release – Ensure accessibility (parent chain): If this option is selected all higher level nodes are also released, starting from the selected object, which were previously never released. This selection is useful, for example, if a new page has been created in the Page Store. With the creation of the new page, not only the release status of the new page changes but also that of the parent node ("Folder"). Both are "not released" (for an example, see Chapter 12.4.1 page 437).

If this option is selected, then, starting from the selected object, released all the parent nodes as well, which were previously never released.

Specific release – Release recursive: If this option is selected all lower level nodes are released also, starting from the selected object. This selection is useful, for example, if many pages below a folder in the Page Store have been changed and now all the changes are to be released together (for an example see Chapter 12.4.2

page 439)

Specific release – Release dependent objects: If this option is selected all objects dependent on the selected object (e.g. a medium used in a picture input component) are released also. A differentiation is made between objects which have never been released to date (**Release new objects only**) and objects which have been re-edited after they had been released (**Release new and changed objects**) (for an example, see Chapter 12.4.3 page 441).

> ! *The specific release option "Release dependent objects" only takes into account the outgoing references of an object. This means the release of a page (Page Store) with this setting release, for example, a reference medium of the page, but not a page reference which refers to this page. Vice-versa, when a page reference is released (Site Store) the referenced page is also released, as this involves an outgoing reference of the object.*

While the release is carried out a detail window appears and shows the progress of the actions.

Click the [View] button after the release has been granted or the [OK] button if the release failed to obtain and check further information in the detail window.
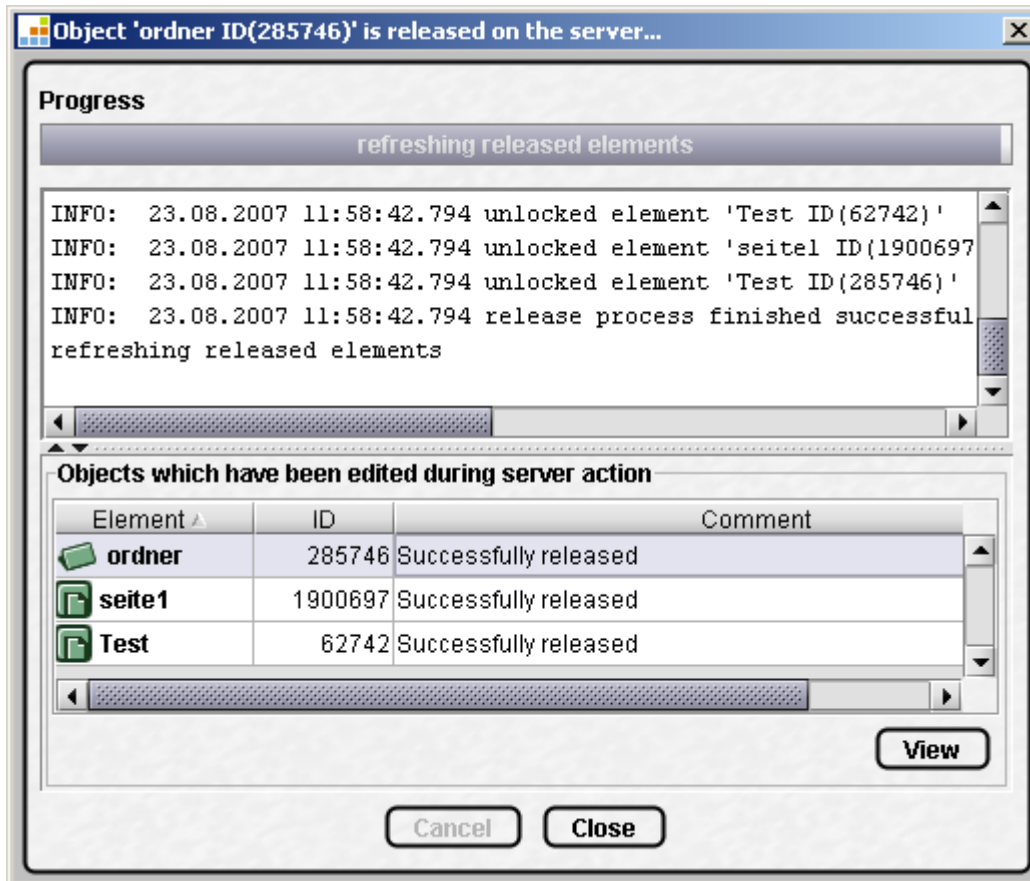
Figure 12-10: Detail window for special release

All objects taken into account in the selected release option (area: "objects which have been edited during the server action") are listed below the progress bar. Apart from the name and ID there is also a comment on each object noting whether the release was successful or not.

View Click this button to display the objects selected from the list in the JavaClient's tree structure.

12.4.1 Example: Specific release - Release parent chain option
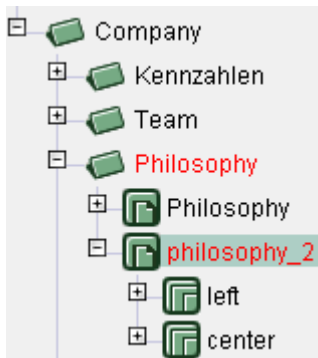
Figure 12-11: Initial situation for Release Parent Chain

If the "Specific release" is now performed on the new page with the "Ensure accessibility (parent chain)" option, not only the new page but also all new folders created in the father chain are released up to the root node of the Page Store.
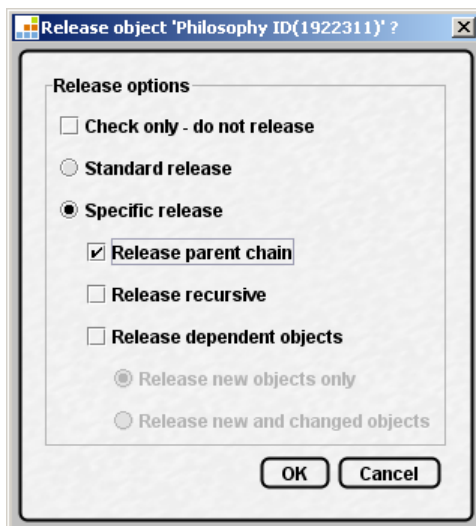
Figure 12-12: "Release father chain" option

According to the settings in the project configuration this dialog enables you to assign a comment, too (see Figure 12-9).

Result of the release:

Figure 12-13: Result of the release

12.4.2 Example: Specific release – Release recursive

Figure 12-14: Initial situation for Release Recursive

If the "Specific Release" is now executed for the higher-level folder with the "Release Recursive" option, all lower-level pages of the folder are released also.
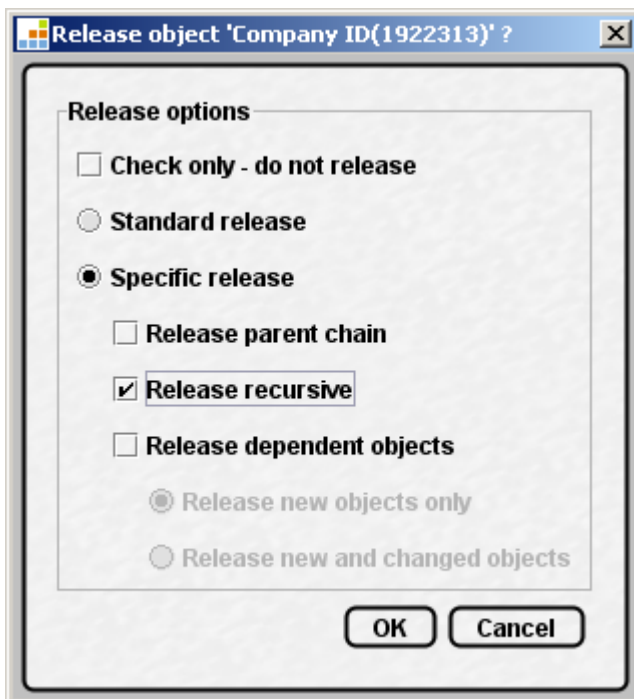
Figure 12-15: "Release Recursive" option

According to the settings in the project configuration this dialog enables you to assign a comment, too (see Figure 12-9).

Result of the release:

Objects which have been edited during server action		
Element △	ID	Comment
Philosophy	1672274	Successfully released
Philosophy	1672293	Successfully released
philosophy_2	1675075	Successfully released
Team	1672205	Successfully released
Company	1672295	Successfully released

Figure 12-16: Result of the release

12.4.3 Example: Specific release – Release dependent objects
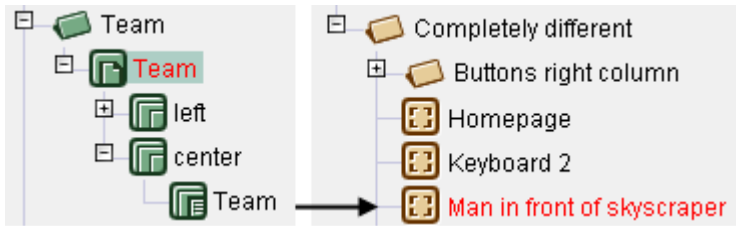
Figure 12-17: Initial situation for Release Dependent Objects

If a medium is changed (or re-uploaded) via the picture input component of a page, this change affects the release status of both the page and the medium.

If the "Specific Release" with "Release Dependent Objects" option is now performed on the page in the Page Store, both the page and the dependent medium are released. The page reference in the Site Store, which this page references, is not released, as from the view of the page (Page Store) it is not an outgoing reference.
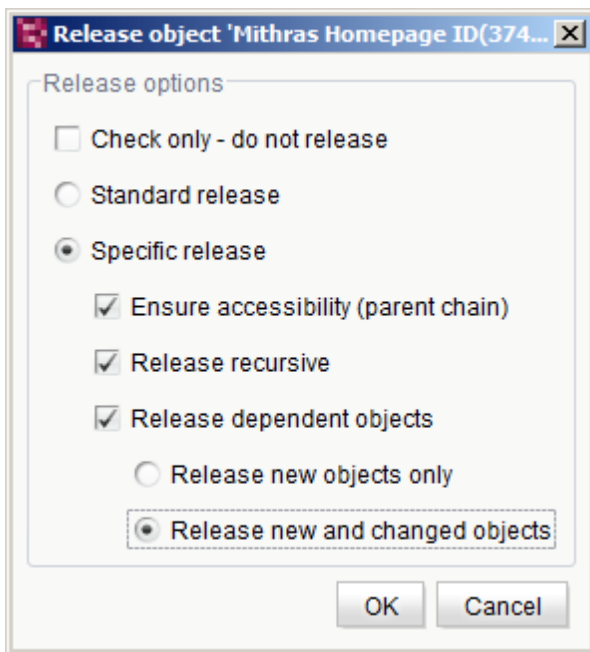
Figure 12-18: Option "Release dependent objects"

> ! *According to the settings in the project configuration this dialog enables you to assign a comment, too (see Figure 12-9).*

Result of the release:

Objects which have been edited during server action		
Element ▲	ID	Comment
🔲 Man in front of skys...	1671617	Successfully released
🔳 Team	1672205	Successfully released

Figure 12-19: Result of the release

> ❗ *The "Release dependent objects" option does not include all objects displayed in the reference graphs under "Display dependencies".*
> *Further documentation on the release of dependent objects is given in the "FirstSpirit Manual for Developers (Part 1: Basics)".*

13 Permissions in the FirstSpirit JavaClient

The mechanisms provided by FirstSpirit for assigning and checking permissions are described in this chapter and their specific use is outlined.

A concrete differentiation must be made between permissions which are valid for a user of FirstSpirit, for example for an editor (editorial permissions) and permissions defined for a user of the page generated with FirstSpirit (user permissions).

FirstSpirit differentiates between the following permissions:

- **Editorial permissions:** These are permissions valid for a user of FirstSpirit. These permissions are initially issued by assigning the user to a project group and can be further specified by authorised persons within the FirstSpirit JavaClient (see Chapter 13.1, page 443).

- **Permissions to execute workflows:** These are a special type of editorial permissions which refer to the workflows within a project only. The permissions to execute workflows are issued parallel to the editorial permissions. (The issue of these permissions is documented in the Developer's Manual, Chapter "4.7 Permissions system for workflows" (see Chapter 13.2, page 462).

- **User permissions:** Are permissions valid for the "user" of the site generated with FirstSpirit. User permissions are always linked with the personalisation system used (see Chapter 13.3, page 471).

13.1 Editorial permissions

Permission management in FirstSpirit enables the allocation of permissions to users and groups. All permissions are issued project-related, this means the permissions only affect the project for which they were defined.

Permissions are always issued in the FirstSpirit JavaClient. Here all areas of the project can be assigned permissions for specific groups of users (see Chapter 13.1.2). Detailed permissions can be assigned for each object, for example an individual page in the Page Store. However, FirstSpirit also enables hierarchical inheritance of permissions within the individual stores (see Chapter 13.1.3).

Initially access to a project is granted by assigning a user to the standard project group "Everyone". This assignment occurs automatically when the project administrator adds a new user to the project using the administrator console.

Advanced editorial permissions can now either be granted directly to the project user or they can be allocated to the project user through the assignment to a project group with the corresponding permissions (see Chapter 13.1.1).

The permissions to be assigned are explained in greater detail in Chapter 13.1.4. All permissions are assigned separately and can be combined with each other in any way required. For example, if a user is only issued the "Visible" permission and "Change Permissions", they have no access to the content of the project, but they could take on the task of permissions management for all other users.

The assignment of editorial permissions only affects projects for which evaluation of editorial permissions has been defined. Otherwise the following message is displayed:
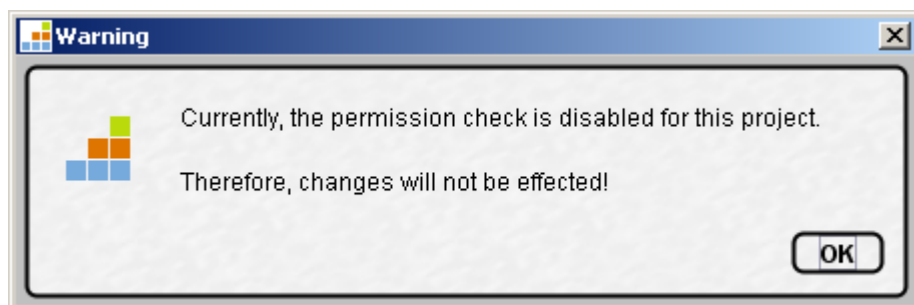
Figure 13-1: Project does not evaluate editorial permissions

> ! *Irrespective of which permissions are defined a project administrator always has at least the permissions: "Visible", "Read" and "Change Permissions".*

13.1.1 Editorial permissions through group membership

A simple and clear option for granting editorial permissions to a user is to assign them to a group. In this case the editorial permissions are defined for the group only and are then valid for all members of the group.

Group assignment can highly simplify the issue and maintenance of editorial permissions. For example, if a certain area of the project is not to be changed by a set of editors and the set of editors changes occasionally, two groups with different permissions are required. The "Editors A" group is issued the permission to "Change" the area, the "Editors B" group is not. By assigning the editors to one of the groups, the appropriate permissions can be set or withdrawn without requiring

any modification of the permission definition in the FirstSpirit JavaClient.

After they have been added to a project each project user is automatically a member of the project group "Everyone". Through this membership the users receive initial access rights (at least the "Visible" permission) to the project. Users can be given advanced or extended access rights by assigning them to further project groups (for example "Editors" or "Administrators").

FirstSpirit differentiates between users who were manually created on the FirstSpirit server and those who were automatically imported from an external system. These imported users can already be members of an external group (group from an external system) and through this group automatically receive advanced access permissions.

> **!** *If a user belongs to several groups with different permissions or if the permissions definitions for one group the user belongs to and the permissions directly assigned to the user contradict each other, a permission is deemed to have been granted if it was granted in one of the settings.*

If necessary, permissions can also be assigned directly to users. In this case the permissions are assigned analogous to the permissions assignment for a group in the FirstSpirit JavaClient (see Chapter 13.1.2).

13.1.2 Permission assignment in the FirstSpirit JavaClient

Editorial permissions for groups and users are defined within the stores in the FirstSpirit JavaClient.

Use the tree view in the left-hand part of the window to select a sub-area of the project for which permissions are to be defined. The highest object to which the changes are to apply is always selected. The permissions set here are inheritable, i.e. they are passed on to all the objects below the selected object (see Chapter 13.1.3 page 452).

Figure 13-2: Context menu - Change permissions

The context menu is now opened at the selected object. Click "Extras / Change

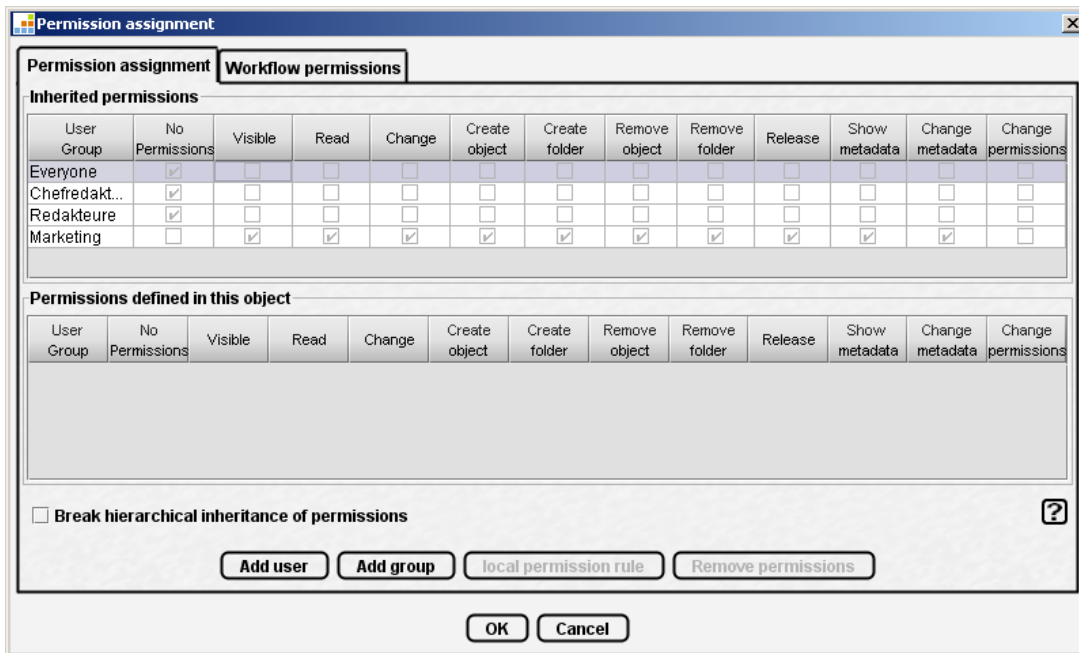permissions" to open the "Permission Assignment" dialog:

Figure 13-3: Permission Assignment dialog - "Permission Assignment" tab

The "Permission Assignment" dialog consists of two tabs. The "Permission Assignment" tab is for allocating editorial permissions, the second "Workflow Permissions" tab is for allocating permissions to execute workflows and is explained in Chapter 13.2 page 462.

Clicking on [User Group] with the right mouse button you can choose, if login, name or name and first name of the users are to be displayed.

[?] Click the question mark to open the Online Help.

The inherited permissions on this node are shown in the top part of the "Permission assignment" tab. **In FirstSpirit Version 4.2R2 and higher** this part is not available at root node level of the stores. The bottom area "Permissions defined in this object" is initially empty as permissions have not yet been defined for this node.

> [!] *For more clearness the entries in the lists "Inherited permissions" and "Permissions defined in this object" are sorted automatically in alphabetical order from FirstSpirit version. First groups and then users are shown.*

If, in addition to the inherited permissions, further permissions are defined for a

group or a user, the corresponding group (see Chapter 13.1.2.3 page 448) or user (see Chapter 13.1.2.4 page 451) must be added to permission definition first.

The "Break hierarchical inheritance of permissions" checkbox must be selected if the permission definition for a group or a user is to be changed. Then a question appears, whether the inherited permissions are to be adopted or not (see Chapter 13.1.2.1 page 447).

13.1.2.1   Adopt ingerited permissions for a node

Figure 13-4: Adopt inherited permissions

No If the dialog is confirmed with "No" the permissions definition for this node starts with deactivated permissions. In this case the permissions set to date for a group or user at a higher level node are not adopted.

Figure 13-5: Example of permission definition without adoption of the inherited permissions

Yes If the dialog is confirmed with "Yes" the inherited permissions are adopted as a default setting:

Figure 13-6: Permissions definition with adoption of the inherited permissions

Irrespective of whether inherited permissions are adopted or not, the permissions for the selected node can now be redefined in the bottom part of the window (see Chapter 13.1.2.2).

The inherited permissions for a group or a user can be quickly adopted by pressing the define as "local permissions rule" button (see Chapter 13.1.2.6 page 452).

13.1.2.2 Define permissions for an existing group/user

A permission for a group or user can be activated or deactivated by simply clicking the relevant checkbox in the "Defined permissions in this object" area.

OK All changes in the "Permissions Assignment" dialog are saved if the changed permissions are confirmed by clicking "OK".

Cancel Click to cancel the assignment of permissions. Already changed or deleted permission definitions are not saved.

13.1.2.3 Define permissions for a new group

If the permissions of an object are to be defined for a new group the required group must be added using the button in the bottom part of the Permissions Assignment dialog window.

Add group Click this button to open the "Groups" dialog with a list of all project groups:
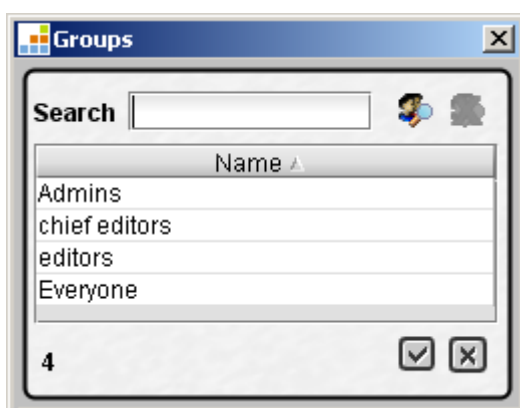

Figure 13-7: Groups

A project group can be selected from the list. Click the ☑ button to add the selected

group to the "Permissions Assignment" dialog where it is included in the "Defined permissions in this object" area (see Figure 13-8).

User Group	No Permissions	Visible	Read	Change	Create object	Create folder	Remove object	Remove folder	Release	Show metadata	Change metadata	Change permissions
Everyone	✔	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Chefred...	✔	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Redakte...	✔	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Marketing	☐	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	☐

Figure 13-8: New group added

The permissions for the new group or user added can now be defined (see Chapter 13.1.2.2 page 448).

> **!** *If a group is added which already has inherited permissions for this node, any additionally assigned permissions are added to the inherited permissions. If permissions are withdrawn from the group but the inherited permissions are not deactivated (see Chapter 13.1.2.1 page 447) the permissions are still deemed to be granted – i.e. are still valid.*

If the **"Everyone" group** is added, the inheritance hierarchy is interrupted and all permissions, apart from the permissions for the "Everyone" group and permissions already assigned for the node concerned in the "Permissions defined in this object" area are withdrawn. After adding the "Everyone" group, the following configuration
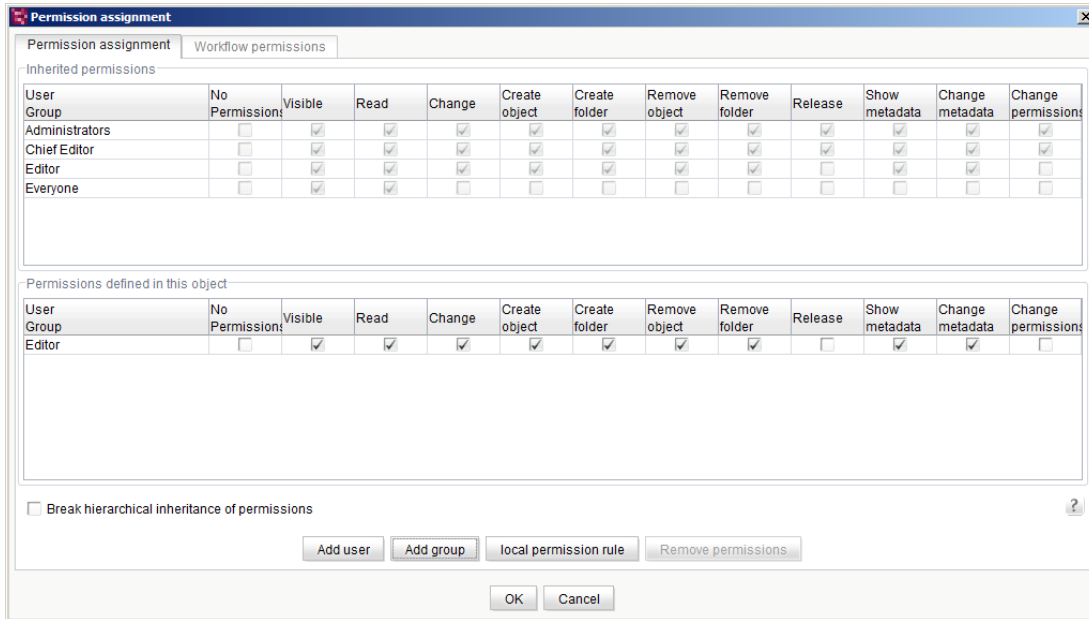
Abbildung 13-9: Add "Everyone" group – Initial configuration

becomes
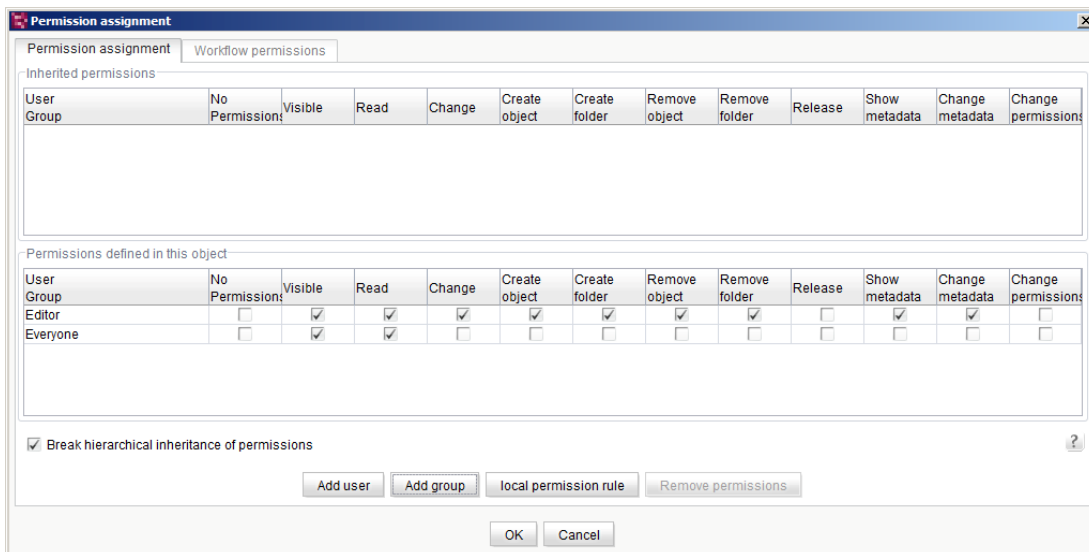
Abbildung 13-10: "Everyone" group added

If other groups are selected at the same time as "Everyone", the permissions of these groups are also adopted.

In this case, **in FirstSpirit Version 4.2R4 and higher** a corresponding message is displayed. If it is confirmed with "Yes", the response is as described above, if "No" or "Cancel" is selected, all selected groups except for "Everyone" are added. Therefore,

if only "Everyone" was selected, no change is made if "No" or "Cancel" is clicked.

13.1.2.4 Define permissions for a new user

If the permissions of an object are to be defined for a new user the required user must be added using the button in the bottom part of the Permissions Assignment dialog window.

Add user Click this button to open the "Add User" dialog with a list of all project users:
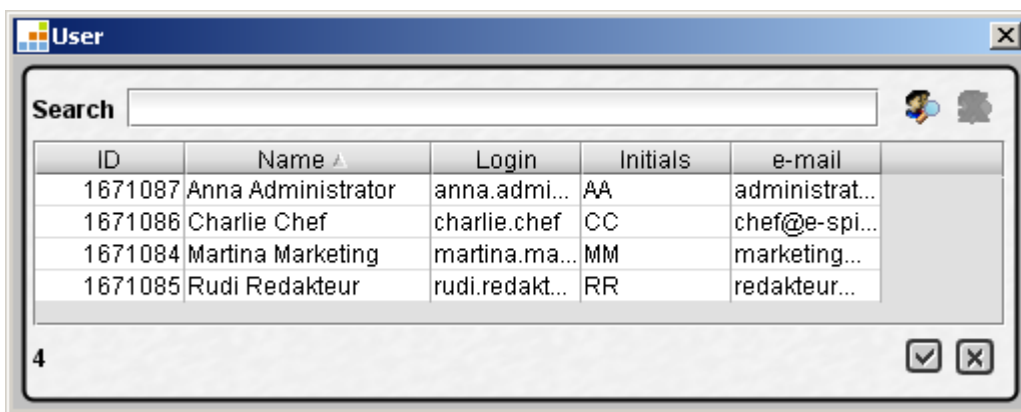
Figure 13-11: Add user

A project user can be selected from the list. Click the ☑ button to add the selected user to the "Permissions Assignment" dialog where they are included in the "Defined permissions in this object" area (see Figure 13-8).

Figure 13-12: New user added

Permissions can now be defined for the new user added (see Chapter 13.1.2.2 page 448).

> ![!] *If a user is added who already has inherited permissions for this node, any additionally assigned permissions are added to their inherited permissions. If permissions are withdrawn from the user but their inherited permissions are not deactivated (see Chapter 13.1.2.1 page 447) the permissions are still deemed to be granted – i.e. are still valid.*

13.1.2.5 Delete permissions defined for this object

The permission definition of a group or user displayed in the "Defined permissions in this object" area can be deleted again. This is done by selecting the group or user to be deleted in the "Defined permissions in this object" area.

Remove permissions Click this button to remove the permissions defined for the selected group or user.

13.1.2.6 Define local permissions rule

local permission rule This button can be used to quickly copy the inherited permissions of a group or user into the "Defined permissions in this object" area where they can then be modified. To this end the user / group is selected in the upper part of the window and then the button is pressed. Local changes to the permission definition can now be made in the bottom part of the window.

> ![!] *Even if the "Local Permission Rule" is used the inheritance still has to be noted and observed (see Chapter 13.1.2.1 page 447).*

13.1.3 Inheritance of editorial permissions within the project

Permissions are assigned in the FirstSpirit JavaClient using the context menu for the objects within the individual stores (see Figure 13-2). This applies to the Page Store, Site Store, Media Store and the global pages in the Global Store. The permissions always apply to the object in the tree for which they were defined and are inherited by all objects at a lower level than this object within the tree structure. Objects or nodes in the tree can be folders, pages, menu levels, page references or media. Permissions cannot be defined at section level. Sections can only exist within the content of a page and therefore inherit their permissions from the higher-level page.

If you want to differ from the high-level permissions in lower-level areas, define new permissions in the required places (see Chapter 13.1.2.1 page 447). However, the assignment permissions at the highest level in the respective store are usually sufficient as these settings are automatically passed on to all other objects in this store.

All nodes in the tree structure at which permissions have been explicitly assigned are denoted by a red dot (see Figure 13-14), from FirstSpirit version 4.1 (look & feel "LightGray") with the symbol 👤. The corresponding permissions symbols only appear in the tree view if the "Show Symbols" setting was activated in the "Extras" menu, from FirstSpirit 4.2 in the "View" menu (see Figure 13-13).
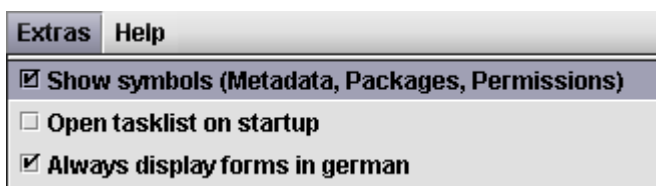
Figure 13-13: Show symbols for the permission definition

For the initial assignment of permissions within a project it is advisable to set the permissions in all stores at the level of the store root and to then redefine them if necessary in the required lower-level objects.
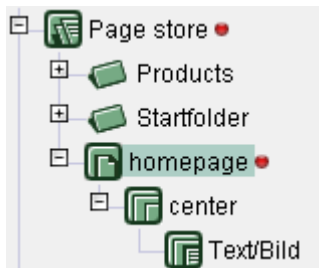
Figure 13-14: symbols for the permission definition

> ❗ *If permissions are withdrawn from a group or user which were defined as issued via inheritance AND if the "Hierarchical Inheritance of the Permissions" is NOT interrupted the withdrawn permissions are nevertheless deemed to have been granted, i.e. they are still valid.*

13.1.4 Possible editorial permissions

User Group	No Permissions	Visible	Read	Change	Create object	Create folder	Remove object	Remove folder	Release	Show metadata	Change metadata	Change permissions
Marketing	☐	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	☐

Figure 13-15: Editorial permissions

13.1.4.1 Permission: Visible

If the "Visible" permission is granted the user can see the tree structure in the left-hand part of the JavaClient window.

If this permission only is granted the right-hand editing area remains deactivated (highlighted in grey) and cannot be edited.

> *The editor needs at least the permission visible for the area page or section templates in the Template Store to be able to create a new page or section in the Page Store.*

13.1.4.2 Permission: Read

If the "Read" permission is granted the content of the object is displayed in the right-hand editing window of the JavaClient. If the object is, for example, a Page Store page the content of the page and all of lower-level sections is displayed.

The "Read" permission does NOT allow the user to change this content! If the "Read" permission only is granted it is not possible to activate the edit mode of the pages, sections, page references and media.

The "Read" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.4.3 The permissions "Visible" and "Read" from V4.1

> *This functionality is released only from FirstSpirit version 4.1. For this reason the screenshots are shown in the new Look & Feel "LightGray". In the Look & Feel "Classic" the representation can differ slightly.*

The security model of FirstSpirit from Version 4.1 differentiates between the **editorial** content of a FirstSpirit object and the **internal** project information of an object (e.g. ID, UID, reference name, display name). Access to this information can be controlled using the "Visible" and "Read" permissions.

Visible: The "Visible" permission is a pure display filter which prevents the **content** protected in this way from being displayed in the FirstSpirit JavaClient. **Information** which is protected by the "Visible" permission is returned by the API and is suitably handled retrospectively, for example, it is hidden. If the "Visible" permission only has been issued, the internal project information (ID, UID, reference name, display name of an object) is displayed to the user, but not the actual content of an object (e.g. the editorial content of a page), which is protected via the "Read" permission.

Read: The decisive difference between the two permissions is that "Read" is an access protection which prevents the data or content of the object protected in this way from being accessed. Therefore, access to objects via the FirstSpirit Access API which are protected by the "Read" permission ("Read" permission is withdrawn), immediately generates a security exception. If the "Read" permission has been issued the editorial content of an object is also displayed to the user.

Exemplary comparison:

User Group	No Permissions	Visible	Read
Everyone	☑	☐	☐

If the "Visible" permission for an object has been withdrawn (corresponds to **No permissions**), it is not displayed in the project structure (tree view) and cannot be selected within the corresponding dialog. If objects for which the user does not have "Visible" (and "Read") permission are referenced in an input component, neither the name nor the content of the referenced object is displayed in the component:
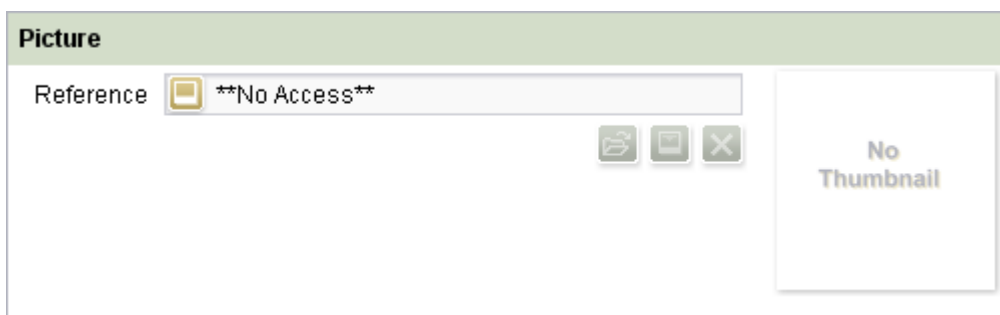
Figure 13-16: "Visible" permission has been withdrawn for the medium

User Group	No Permissions	Visible	Read
Everyone	☐	☑	☐

If the **Visible** permission is set for an object it is displayed in the project structure (tree view), however, the editorial content, for example, of a medium is not displayed. This content is protected from access as the "Read" permission has not been issued for the object:
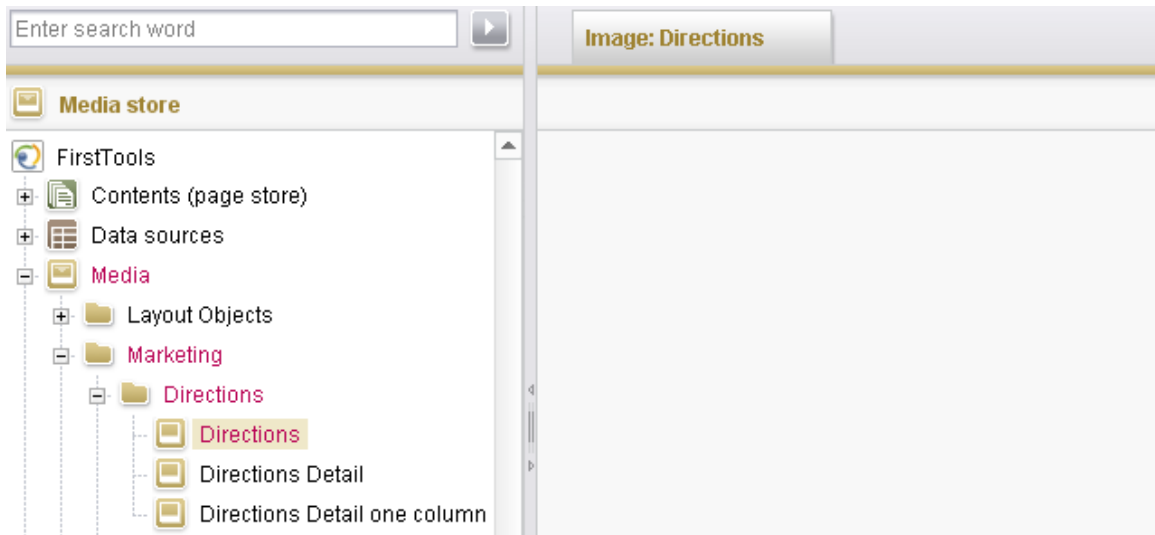
Figure 13-17: "Visible" permission has been set for the medium

The same also applies to the display of protected content in input components. For example, if a medium to which the user solely has "Visible" permission is selected here, the user can select the medium but the content is protected against access ("No Thumbnail"):

Figure 13-18: "Visible" permission has been set for the medium – Display in the input component

User Group	No Permissions	Visible	Read
Everyone	☐	☑	☑

If the **Visible** permission and the **Read** permission have been set for an object, it is displayed both in the project structure

(tree view) and with its editorial content.
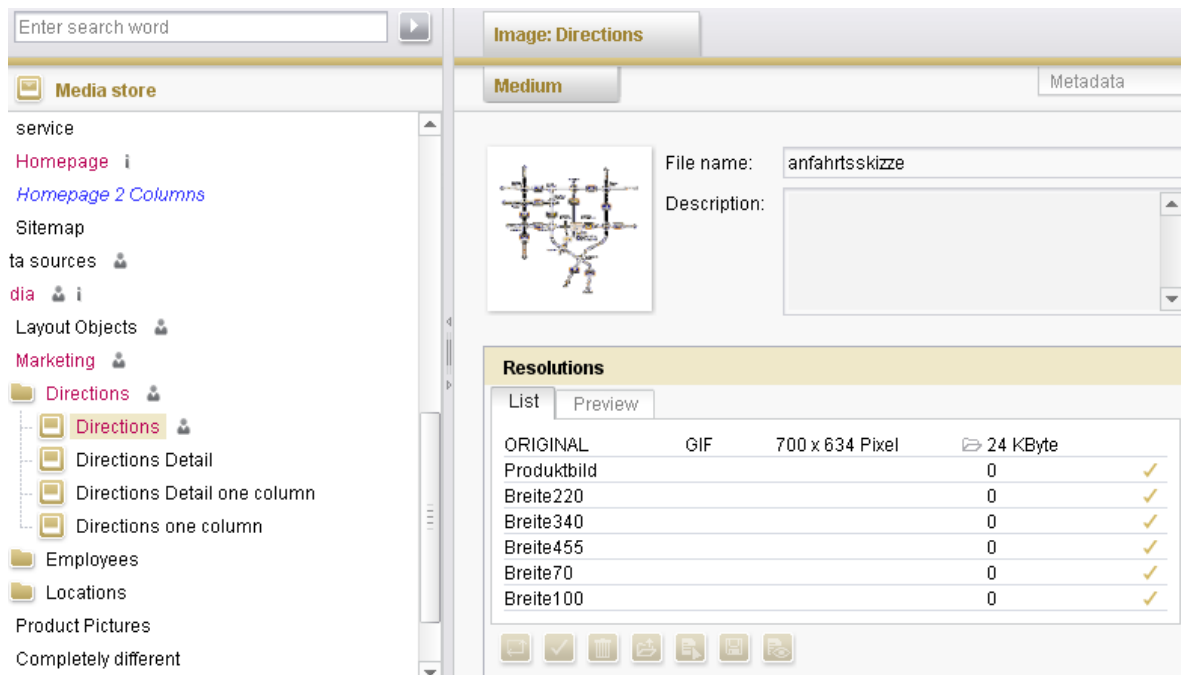
Figure 13-19: The "Visible" and "Read" permissions have been set for the medium

The same also applies to the display of protected content in input components. If, for example, a medium is to be selected here to which the user has the "Visible" permission and the "Read" permission, they can select the medium; in addition, the editorial content is displayed to them, e.g. in the form of a picture preview:

Figure 13-20: The "Visible" and "Read" permissions have been set for the medium

13.1.4.4 Permission: Change

If the "Change" permission is granted the user can make changes to the object and to the object's content. The "Change" permission includes:

- Renaming the object

- Setting object in Edit Mode (see Chapter 3.2.6 page 92).

- Change contents of the object.

- In the Page Store this permission also relates to:

  o Deleting sections
  o Adding sections
  o Copying sections

The "Change" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.4.5 Permission: Create object

If the "Create Object" permission is granted the user can create the following objects:

- In the Page Store: Pages and sections

- In the Content Store: Add rows to a data source, to 4.2R4 including: Set filter

- In the Media Store: Media

- In the Site Store: Page references and document groups

It is not possible to change the existing folder structure or menu levels. All objects can only be inserted in the existing structure.

> ![!] *Sections can only exist in the content area of a page. The user therefore needs the "Change" permission and not the "Create Object" permission to be able to create new sections within a page.*

> ![!] *The user needs at the same time the permission "Visible" in the Template Store for the complete path to the desired page template to be able to create a new page.*

The "Create Object" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.4.6 Permission: Create folder

If the "Create Folder" permission is granted the user can insert new folders in the

store structures.

- In the Page Store: Create new folders

- In the Content Store: Create new folders and add new data sources, from 4.2R4: set filter.

- In the Media Store: Create new folders

- In the Site Store: Create new menu levels

The "Create Folder" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.4.7 Permission: Remove object

If the "Remove Object" permission is granted the user can delete the following objects:

- In the Page Store: Pages with sections

- In the Content Store: Entries of a data source.

- In the Media Store: Media

- In the Site Store: Page references

The "Remove Object" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

> *Deleted objects can be restored using the "Delete" icon (see Chapter 3.2.9 page 96). The editor does not require any permissions to "Create Object" for this as in this case it does not involve a new object.*

13.1.4.8 Permission: Remove folder

If the "Remove Folder" permission is granted the user can remove folders in the stores. The objects to which this permission refers are identical with the objects which can be created using the "Create Folder" permission (see Chapter 13.1.4.6 page 458).

If a folder is deleted all its lower-level objects are automatically removed too. For example, if folders in the Page Store are removed all lower-level folders, pages and sections are removed too.

If there are elements below a folder for which the user does not have permission to delete these elements are retained.

The "Remove Object" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.4.9 Permission: Release

If the "Release" permission is granted the user can "release" changed objects (for info on the release of objects, see also Chapter 12.3 page 425).

In the FirstSpirit JavaClient objects are released using a workflow. Within the workflow the object is converted from "not released" status to "released" status. The permission to "release" relates to precisely this procedure, the conversion of the object into the "Release status". The right to execute the individual steps of the "Release" workflow is arranged by the granting of permissions to execute workflows (see Chapter 13.2 page 462).

The "Release" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.4.10 Permission: Show metadata

Metadata can be defined for each object, providing working with metadata has been configured for a project. The metadata can be maintained using forms in precisely the same way as other project contents and are different in a project-specific way (see Chapter 11.3 page 317). A special form of metadata is, for example, user permissions, whose maintenance via metadata is explained in Chapter 13.3.2 (page 472).

If the "Show Metadata" permission is granted, already entered metadata is displayed to the user (for example, already assigned user permissions).

The "Show Metadata" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.4.11 Permission: Change metadata

If the "Change Metadata" right is granted the user can make changes to the content of the metadata.

The "Change Metadata" permission has dependences on other editorial permissions

(see Chapter 13.1.5 page 461).

13.1.4.12 Permission: Change permissions

If the "Change Permissions" permission is granted the user can execute the permissions assignment for groups and users described in this chapter. It is advisable to only grant this permission to persons who are assigned the role of a project administrator.

The "Change Permissions" permission has dependences on other editorial permissions (see Chapter 13.1.5 page 461).

13.1.5 Dependences between editorial permissions

Certain permissions can only be usefully granted if the user or group also have other permissions. For example, the "Read" permission can only be usefully granted if the "Visible" permission was issued simultaneously. Otherwise the user could theoretically read the contents but practically it is not possible for them to select these contents via the tree view. The "Read" permission is therefore dependent on the "Visible" permission.

FirstSpirit supports the assignment of such dependent permissions. If a group or user is granted a permission which is dependent on another editorial permission, when the editorial permission is granted the dependent permission is automatically granted too.

Dependences exist between:

Visible:	No dependences.
Read:	Dependence on the "Visible" permission.
Change:	Dependence on the permissions: ▪ "Visible" ▪ "Read"
Create object:	Dependence on the "Visible" permission.
Create folder:	Dependence on the "Visible" permission.
Remove object:	Dependence on the "Visible" permission.
Remove folder:	Dependence on the "Visible" permission.

Release:	Dependence on the permissions: ▪ "Visible" ▪ "Read"
Show metadata:	Dependence on the permissions: ▪ "Visible" ▪ "Read"
Change metadata:	Dependence on the permissions: ▪ "Visible" ▪ "Read" ▪ "Change"
Change permissions:	Dependence on the permissions: ▪ "Visible" ▪ "Read" ▪ "Change"

13.2   Permissions to execute workflows

Permissions to execute workflows are a special type of editorial permissions which only relate to the workflows within a project (for further information on workflows see Chapter 12, page 422). The permissions for executing workflows are assigned parallel to the editorial permissions for groups and users within the stores in the FirstSpirit JavaClient (cf. Chapter 13.1.1, page 444).

The permissions are assigned analogous to the assignment of user permissions using the "Permissions Assignment" dialog (see Chapter 13.1.2, page 445).

13.2.1   Permission assignment in the FirstSpirit JavaClient

The "Permission Assignment" dialog consists of two tabs. The "Permission Assignment" tab is for the allocation of editorial permissions, the second "Workflow Permissions" tab is used to allocate permissions to execute workflows.
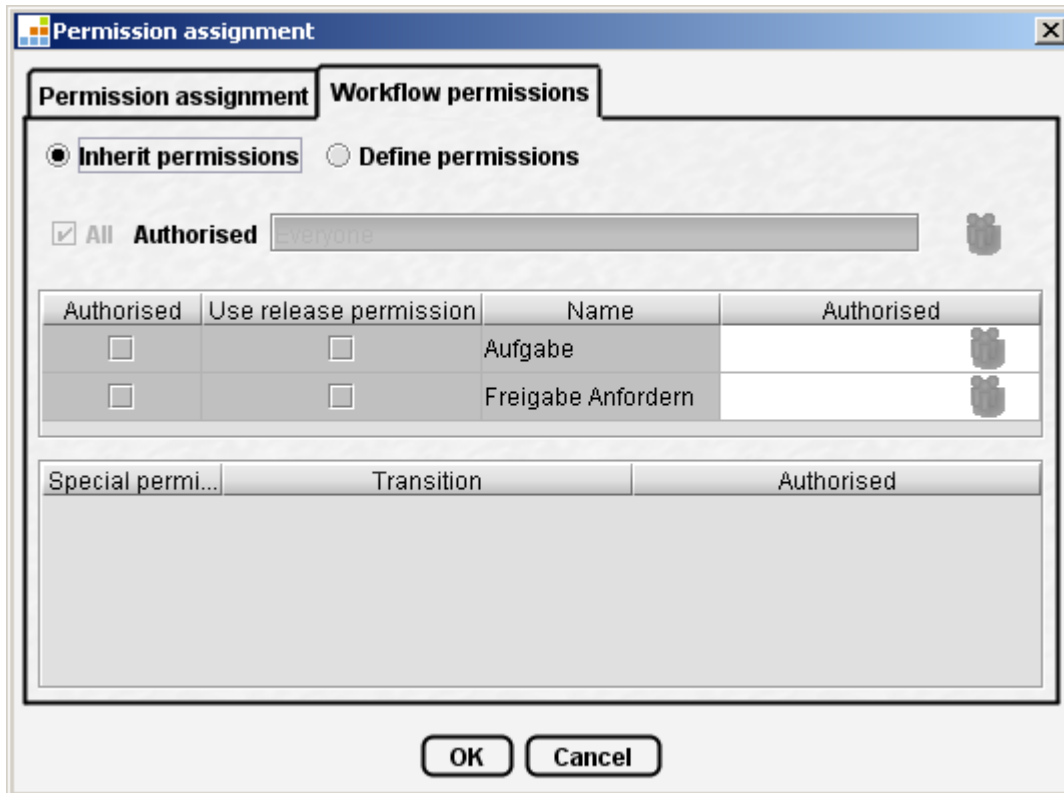
Figure 13-21: Permission Assignment dialog - "Workflow Permissions tab"

Users authorised to start all the workflows in the project can be defined in the top part of the "Workflow Permissions" tab. The bottom part of the tab is for allocating permissions and special permissions for individual workflows.

Inherit permissions: The "Inherit Permissions" radiobutton is selected as a default (exception: root nodes). With this setting the "Workflow Permissions" are inherited from a high-level node (see Chapter 13.1.3 page 452).

Define permissions: The "Define Permissions" radiobutton must be activated if the permissions definition for the execution of a workflow on the current node is to be changed. Then a question appears, whether the inherited permissions are to be adopted or not (see Chapter13.1.2.1 page 447). Permissions defined on a node are inherited by all lower-level objects (see Chapter 13.1.3 page 452).

Figure 13-22: Adopt inherited permissions

[**No**] If the dialog is confirmed with "No" the permissions definition for this node starts with deactivated permissions (see Figure 13-21). In this case the permissions set to date for a group or user at a higher level node are not adopted.

[**Yes**] If the dialog is confirmed with "Yes" the inherited permissions are adopted as a default setting:

Figure 13-23: Permissions definition with adoption of the inherited permissions

Irrespective of whether inherited permissions are adopted or not, the permissions for the selected node can now be redefined (see Chapter 13.2.2 ff.).

13.2.2   Define permissions for all workflows

To make the permissions assignment as convenient as possible it is possible to simultaneously set the execution permissions for all workflows in the project:

All: If this checkbox is activated the permission to start is defined for all workflows in the project. The table with the list of workflows then becomes disabled and can no longer be edited. In this case the settings made within the table for the individual workflows are not evaluated.

Figure 13-24: Define permissions for all workflows

Authorised: All users and/or groups who are authorised to execute a workflow on the current node are listed in this field (add or delete authorised persons, see Chapter 13.2.4 ff.).

> *The permissions defined here for the execution of workflows are solely related to starting the respective workflow. The permissions for execution of a transition (from one step of the workflow to the next step) are specified either via the Template Developer in the workflow or by assigning special permissions for the individual steps of a workflow (see Chapter 13.2.5 page 469).*

OK All changes in the "Workflow Permissions" dialog are saved if the changed permissions are confirmed by clicking "OK".

Cancel Click to cancel the assignment of permissions. Already changed or deleted permission (permissions) definitions are not saved.

13.2.3 Define permissions for individual workflows

All: If this checkbox is activated the permissions to start the workflows in the project are defined individually. The table with the list of workflows then becomes active and can be separately edited for each workflow.

Authorised	Use release permission	Name	Authorised	
☑	☐	Aufgabe	Everyone	👥
☑	☑	Freigabe Anfordern	Everyone, martina.marketing	👥

Figure 13-25: Define permissions for an individual workflow

Authorised: If this checkbox is activated authorised persons may start this workflow on the currsent node of the tree structure. As soon as the checkbox has been activated the permissions assignment for the current workflow can be defined in detail in the bottom part of the window (see Chapter 13.2.5 page 469).

Use release permissions: If the "Use Release Permissions" checkbox is activated the release permissions defined in the "Permissions Assignment" tab are evaluated for each user (see Chapter 13.1.4.9 page 460).

> ❗ *Contradictions in the permissions definition can occur if the checkbox is not activated. Conflict situations can arise if, for example, a user has no right to release a specific object but is listed as being "authorised" in the standard "Request Release" workflow. In such a case the release would be prevented by the system but the response (no release) is not transparent for the user as the workflow can be passed on as defined up to the "Grant Release" status. If on the other hand the "Use Release Permissions" checkbox is activated the release permissions of the users are evaluated at each workflow transition to the following status. If contradictions are found between the editorial permissions (no permission to release) and the permissions in the workflow (e.g. grant release) these transitions are hidden for the "unauthorised" user. In this case the user can "Request Release", i.e. start the workflow but they can no longer switch the object to the following "Object Released" status. The transition required for this is hidden.*

Name: Unique name of the workflow from the Template Store.

Authorised: All users and/or groups who are authorised to execute a workflow on the current node are listed in this field (add or delete authorised persons, see Chapter 13.2.4 ff.).

FirstSpirit™

[OK] All changes in the "Workflow Permissions" dialog are saved if the changed permissions are confirmed by clicking "OK".

[Cancel] Click to cancel the assignment of permissions. Already changed or deleted permission (rights) definitions are not saved.

13.2.4 Change authorised groups / users

The [icon] icon in the "Authorised" column can be used to change the selection of authorised persons. If the icon is clicked the "Select Groups/Users" dialog opens.
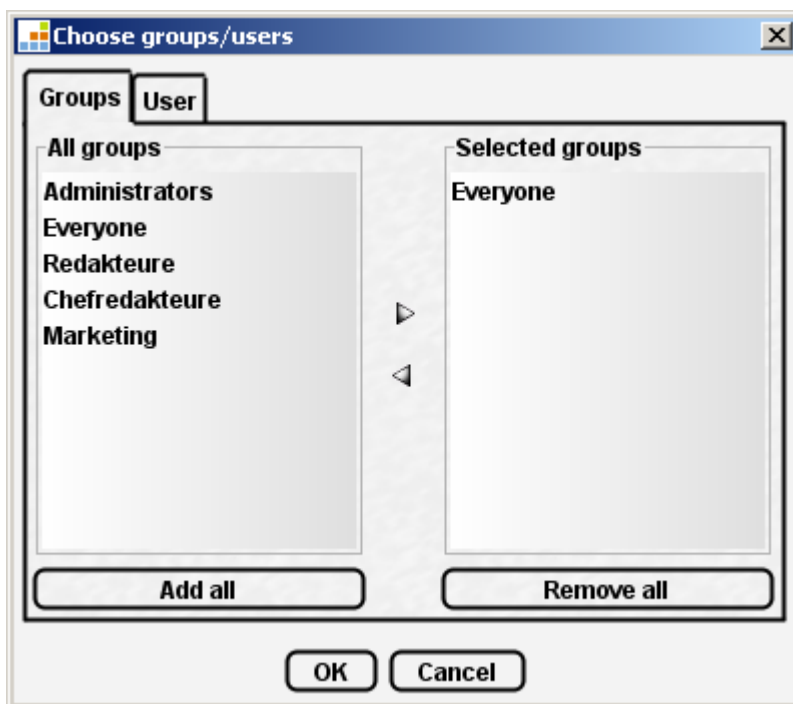
Figure 13-26: Select authorised groups / users

The dialog is divided into two: All the project's users ("Users" tab) and groups ("Groups" tab) are listed in the left-hand side of the window. The groups and / or users already selected in the "allowed" field are displayed in the right-hand side of the window.

The icons can now be used to add a selected group (in the left-hand area) to the selected groups ▷ or to remove a selected group (in the right-hand side) from the "selected groups" area ◁.

The addition and removal of users in the "User" tab is carried out analogous to this.

Add all Click this button to transfer all users and/or groups from the left-hand side of the window into the "Selected Groups" or "Selected Users" area.

Remove all Click this button to remove all users and/or groups from the "Selected Groups" or "Selected Users" area.

OK All changes in the "Select Groups/Users" dialog are saved.

Cancel The dialog is cancelled, changes are not saved.

13.2.5 Permissions assignment for executing the transitions

If permissions have been defined on a node for a workflow (see Chapter 13.2.3 page 466), the individual steps of this workflow are displayed in the bottom part of the "Workflow Permissions" tab. The permissions for executing the workflow can be defined in detail in this area for each step of the process.

> ❗ *Permissions for passing on a workflow defined at this point overwrite the permissions defined for this workflow by the template developer.*

To assign these permissions a workflow is first selected in the upper table. All steps belonging to this workflow are now shown in the bottom table:

Authorised	Use release permission	Name	Authorised	
☑	☐	Aufgabe	Everyone	👥
☑	☑	Freigabe Anfordern	Everyone, martina.marketing	👥

Special permi...	Transition	Authorised	
☐	object changed->request release	Everyone	👥
☐	request	Everyone	👥
☑	check	Chefredakteure	👥
☑	do not grant	Chefredakteure	👥
☐	edit	Everyone	👥
☐	new request	Everyone	👥
☑	grant	Chefredakteure	👥
☑	direct release	Chefredakteure	👥
☐	final		👥

Figure 13-27: Special permissions for the execution of a workflow

Special permissions: If the checkbox is selected, the permissions assigned in the workflow for this transition are ignored *on this node*. Only the users and groups defined in the respective special permission are authorised to execute the transition. If the checkbox is deactivated the permissions for executing the transition defined by the template developer for this workflow are evaluated.

> ❗ *The persons authorised to execute special permissions do not have to be identical with the users or groups who are authorised to start the workflow (see Chapter 13.2.3 page 466).*

Transition: Names of the transitions assigned in the workflow by the template developer. If a name has not been assigned to a transition the names of the source and target of the transition are displayed (cf. Figure 13-27 - first step).

Allowed: All users and/or groups who are authorised to execute this transition on this node are listed in this field (add or delete authorised persons, see Chapter 13.2.4 ff.).

Figure 13-27 shows the assignment of special permissions with the help of the standard "Request Release" workflow. The following steps must be run through to execute the whole workflow (cf. Chapter 12.3 page 425):

1. The workflow must be started if an object has been changed, for example a page of the Page Store. No special permissions have been defined for this. Therefore, the persons for whom starting the workflow was "allowed" (top table) are authorised on the current object.

2. In the next step the release is requested. Permissions were also not defined (on this object) for this transition either. Therefore the permissions specified by the template developer for this workflow transition are evaluated.

3. In the next step the changes to the object must be checked. This transition (on the current object) should only be allowed to be executed by the "chief editors" group. The permissions defined in the workflow are therefore switched off (for the current object).

4. After the changes have been checked the release can be "granted" or "not granted". This decision (on the current object) should also only be allowed to be taken by the "chief editors" group.

5. If the release was "not granted" the change to the object must be re-"edited". As special permissions have not been defined for this transition the permissions from the workflow are again evaluated.

6. A release can then be requested again.

13.3   User permissions

There is strict differentiation between editorial and user permissions in FirstSpirit. Unlike the editorial permissions which relate to processes in the FirstSpirit project, user permissions solely relate to the site generated and published with FirstSpirit and are therefore closely linked with the personalisation system used (e.g. FirstSpirit Personalisation[2]). In many cases a user permission is interpreted as "permission to view an object". However, there are also feasible cases in which apart from "Visible" the "print" and "save" permissions are also relevant. It is usually possible to tell whether a project works with user permissions by the use of a login page for "visitors" to the site.

13.3.1   User permissions through group membership

The editorial permissions (e.g. create, change, delete, release), just like the assignment of these permissions to project groups or project users, are rigidly specified by FirstSpirit (see Chapter 13.1 page 443).

Comparable specification is not possible for user permissions as each FirstSpirit project sets completely different requirements regarding user permissions. Therefore, when user permissions are defined, neither the permissions, i.e. the possible operations on an object, nor the group structures are defined using FirstSpirit. This definition is project-specific, differs for each project and is the responsibility of the project administration and template development (for further documentation, see "FirstSpirit Manual for Developers" and "FirstSpirit Manual for Administrators").

> **!**   *The group structures and user permissions used in the project can therefore be highly different from those described here by way of example.*

In FirstSpirit, user permissions are assigned on the basis of groups. Unlike the groups structures created in FirstSpirit for the assignment of editorial permissions (cf. first section), the groups used here can have a hierarchical structure – i.e. a group can contain several sub-groups - and so reflect the hierarchies within a company.

---

[2] See documentation for the module FirstSpirit Personalisation

The group hierarchy is presented to the editor in the form of a tree view in which the permissions can be configured (see Chapter 13.3.2 page 472).

13.3.2 Permission assignment in the FirstSpirit JavaClient

In the FirstSpirit JavaClient the user permissions are assigned using a special input component. User permissions can be assigned on the basis of a hierarchical group definition using the permission definition component (see Chapter 13.3.1 page 471).

The permission definition component is most frequently used within the scope of the metadata definition but can also be used in other page and section templates and can therefore be used in the Site Store.
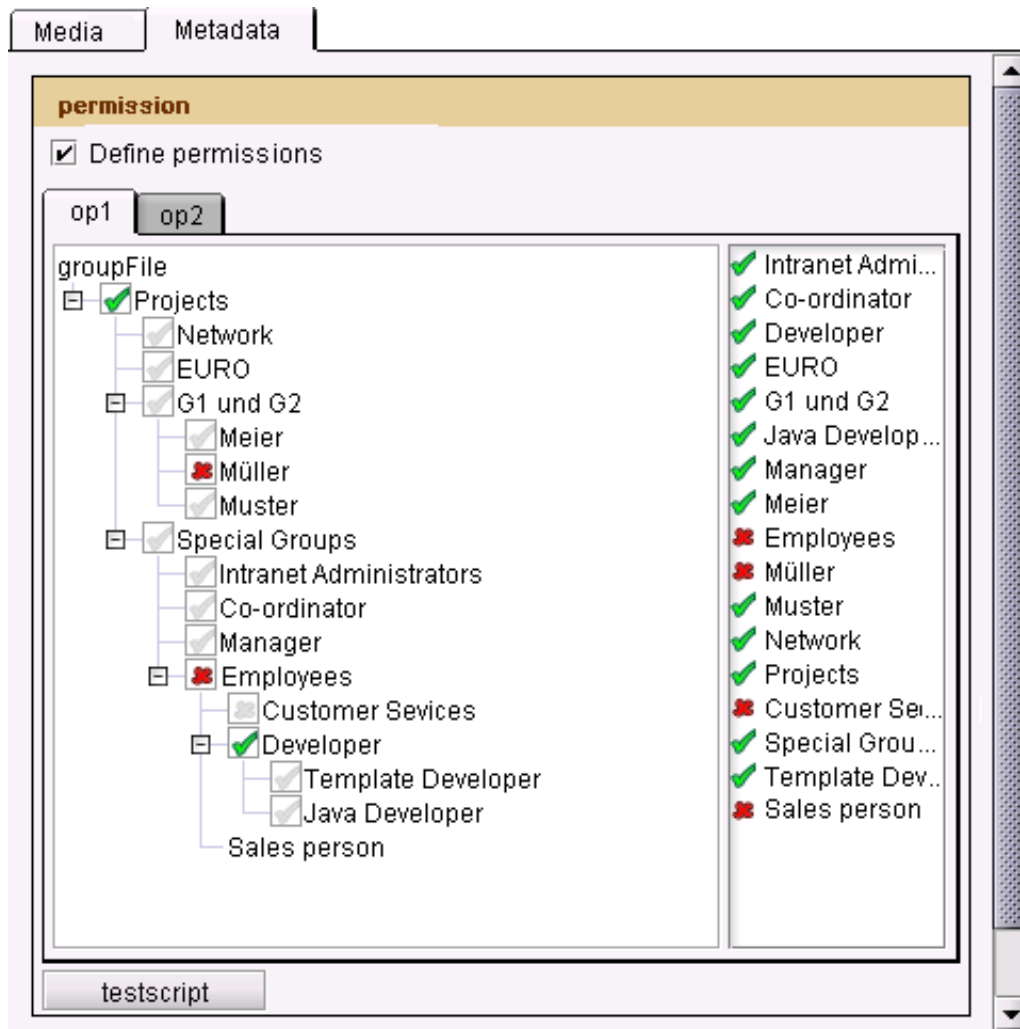

Figure 13-28: Permission definition component

Define permissions: Activate the checkbox to interrupt the inheritance hierarchy of the permission definition. The permissions can then be redefined for the selected

node in the tree structure (and all nodes below it) (see Chapter 13.3.4 page 475).

A tab is displayed in the input component for each user permission (for example "Read") to be defined using the permission definition component (cf. Figure 13-28 "op1" and "op2"). Therefore, it is possible to separately specify for each user permission for which user groups an action is to be explicitly allowed or prohibited.

The individual groups with their subgroups are displayed in a tree structure on the left-hand side of the component. A symbol is displayed in front of each group which indicates which permission is valid for this group and where this right was defined.

✔ = Allowed, defined here

✖ = Prohibited, defined here

✔ = Allowed, defined in a parent

✖ = Prohibited, defined in a parent

The valid permission for each group is displayed in an overview list on the right-hand side of the component.

In Figure 13-28 the permissions are distributed as follows:

▪ An explicit permission is assigned for the "Projects" group, therefore all subgroups of the "Projects" group are also allowed access to the defined permission "op1".

▪ Access to the defined right is explicitly prohibited for the "Müller" group (Path: "Projects" – "G1 and G2" – "Müller").

▪ Equally, an explicit ban is also assigned for the "Employee" group (Path: "Projects" – "Special Groups" – "Employees").

▪ The "Employees" group contains further subgroups for which access to the defined permission is also prohibited by default.

▪ Access to the defined permission is then explicitly allowed again for the "Developer" subgroup (Path: "Projects" – "Special Groups" – "Employees" – "Developer").

▪ Access to the defined permission is also allowed by default for the subgroups of the "Developer" group.

▪ A special rule applies to the "Salespersons" group. This group always has the same permission as the parent node, it is not possible to assign an explicit permission for this group (the field for permission assignment does not exist).

If permissions have not been defined for a selected node, however they have been for a node at a higher level, the definition of the node on which the permissions were defined is displayed. This can be identified by the fact that there is no green dot

behind the name of the selected node in the tree or that – in the Look & Feel "LightGray" from FirstSpirit Version 4.1 – there is no icon i (if no other metadata has been defined) and the tick is missing at "Define Permissions".

Evaluation of the user permissions is explained in the following chapter using an example.

13.3.3 Evaluation of the user permissions

The evaluation of the user permissions (right-hand area) in conjunction with the group hierarchy (left-hand area) is not trivial and should therefore be explained in greater detail using an example:
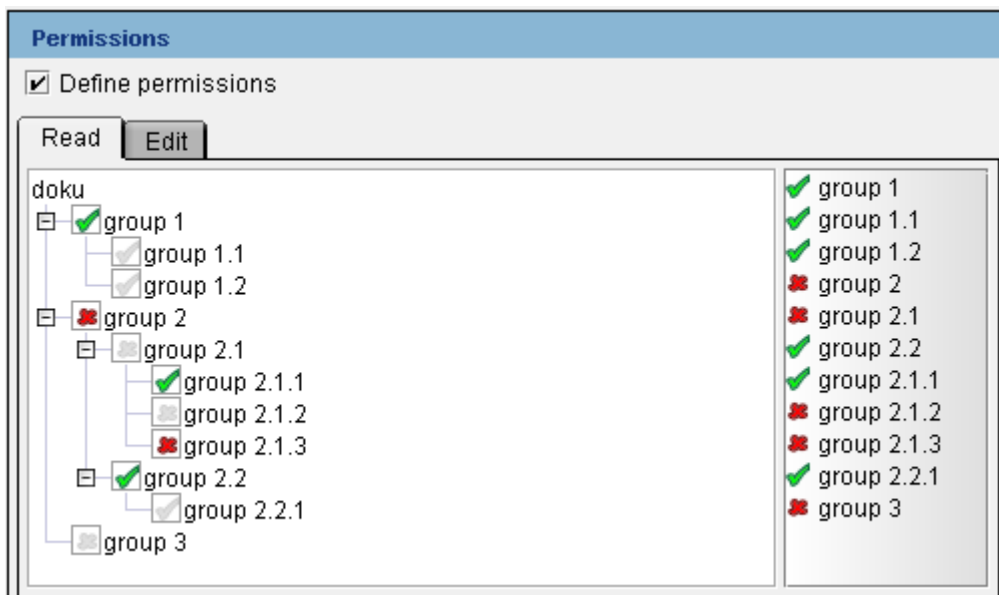

Figure 13-29: Permission definition component example

Access is basically prohibited for all " doku". I.e. this is the default value for all groups for which nothing else is defined (e.g. " group 3") – therefore, if a new group is added at the highest level, "Prohibited" applies.

"Group 1" has an explicit permission through " group 1" which as a default also has an effect on subgroups 1.1 and 1.2. If a new group is added to this level access is allowed as a default.

"Group 2" is also configured – in this case with " everything prohibited". This has a default effect on 2.1 and in 2.2 is explicitly " allowed" again. The permission in 2.2 has an implicit effect on 2.2.1. Therefore, all members of 2.2 are authorised. In node 2.1 access is differentiated still further: Here "group 2.1.1" is explicitly allowed while

"group 2.1.3" is explicitly prohibited. No specification is made for "group 2.1.2", the value therefore depends on the first explicitly configured parent (here: "group 2").

Therefore, if "group 2" is reconfigured from to , " group 2.1.2" changes to " group 2.1.2" (Note: this also applies of course to "group 2.1" too but not to "group 2.1.1" and "group 2.1.3"). Effectively, this configuration results in people who are only in "group 2" do not have any access and people who are in 2.2 or in one of its subgroups or in 2.2.1 are given access. Access to people in groups 2.1.2 and 2.1.3 remains refused. Those in 2.1.3 are even still refused access if 2 is reconfigured to .

The evaluation of the authorisation configuration is displayed on the right-hand side of the input component (see Figure 13-29). The evaluation is issued in that the complete group tree is run through and the evaluation for each node is inserted in an "Allowed" or in a "Prohibited" list. Prerequisite for this is that default assignment takes place at the root level (i.e. "doku").

In general a differentiation is made between nodes on which explicit permissions have been defined and nodes on which no permissions have been defined. If permissions have not been defined on a node the permissions of the parent node apply. If the parent node does not have any defined permissions either the permissions of the first high-level node for which explicit permissions have been defined are adopted. If permissions are not defined in any node, the value is used as a default for all nodes.

13.3.4 Inheritance of user permissions within the project

Apart from group hierarchy, user permissions also have a relation to the tree structure of the FirstSpirit stores, which is also interpreted as hierarchy.

Analogous to the inheritance of editorial permissions (see Chapter 13.1.3 page 452), the user permissions also always apply to the object in the tree for which they were defined and are inherited by all objects at a lower level than this object within the tree structure. This means: If there are no user permissions defined in a tree object the permissions of the parent object apply. Through this inheritance definition is very easy e.g. at the level of a folder, to define the permissions for all pages below it.

The inheritance is defined as "not additive" – this means that a permission definition in an object overwrites all definitions "above it".

> **!** *Initially, metadata is not set for the permission definition component in a project. The respective root nodes should be set for a basic definition.*

> **!** *Contradictions can arise in hierarchical structures if permissions are explicitly defined on a node. For example, if a group's access to a Site Store folder is explicitly prohibited but its access to a lower-level folder is explicitly allowed, these defined permissions contradict each other. The plausibility of the permission assignment is not checked.*

To avoid contradictions, for example with permission definition within the Site Store, the quantity of authorised groups along the tree should only be restricted but never extended as in this case access to a "deeper element in the tree" can only be achieved via the "node above". Therefore, extending the authorisation is pointless as the high-level entry point is missing.

Contradictory permissions assignments within a project can be uncovered using a script. The component supports the linking of scripts which, for example, are executed "on clicking", i.e. immediately on defining a right or permission in the component, or can be called via a button only if a check is explicitly required (example: Button - cf. Figure 13-28).

Checking via a script must be adjusted for the specific project by the template developer!

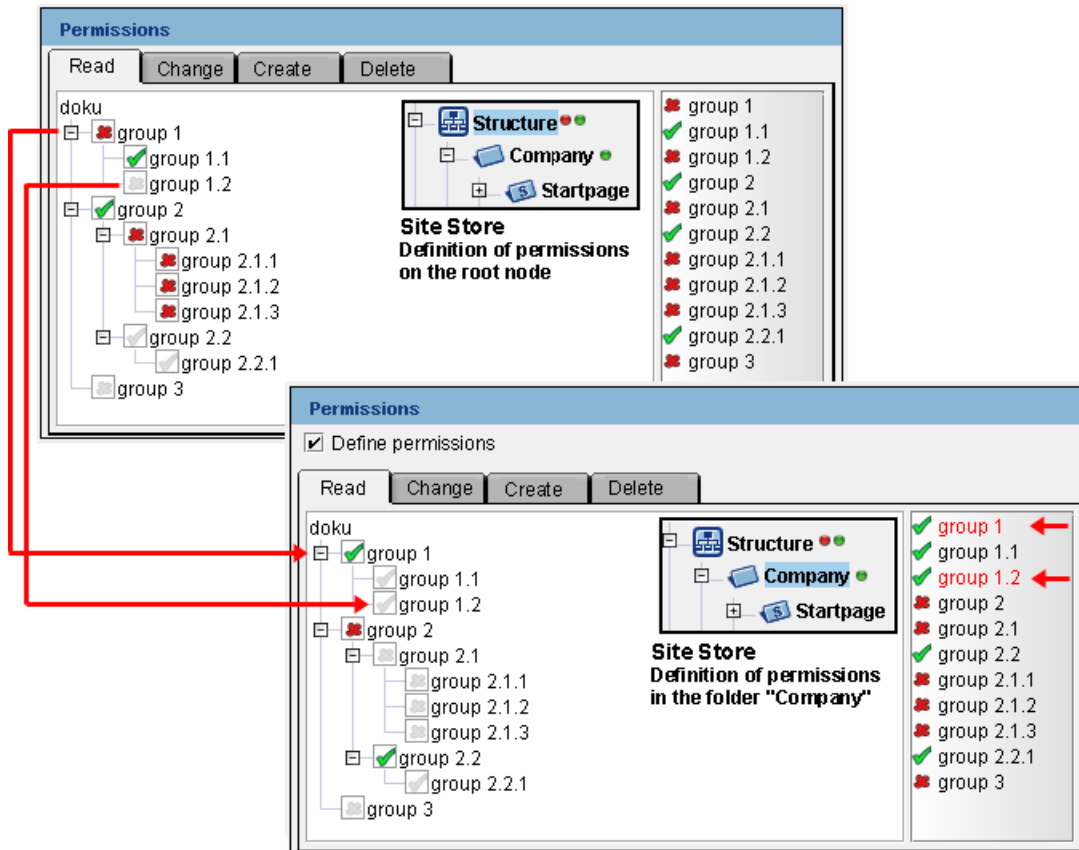Example of contradictory permission assignment within the Site Store:

Figure 13-30: Permission definition with contradiction

For example, if the "Read" permission for "group 1" is explicitly prohibited on a root node of the Site Store, it cannot be explicitly allowed for this group in the lower-level "Company" folder. This permission assignment contradicts the permission definition of the parent node. The same contradiction affects the inherited permissions of "group 1.2". The situation is different for "group 1.1". This is explicitly assigned "Read" permission in the root node and can also retain this permission in the lower-level "Company" folder. It is of course possible to restrict permissions in the lower-level element, as for "group 2" at any time.

13.3.5 Dependencies of user and editorial permissions

In several cases there is a close relationship between editorial permissions and user permissions:

1) In the preview of a page:
    In this case the editor is also the user - here the editorial permission "Visible"

and the user permission "Visible" clash and must be appropriately linked.

2) Data change on the live site:
    In this case the user is also the editor – here, analogous to the above, the user permission "change" and the editorial permission "change" must be appropriately linked.

The link is usually established by an additional login process, i.e. the user logs in as an editor and vice-versa.