



# FirstSpirit™

*Unlock Your Content*

## FirstSpirit™ Manual for Editors (JavaClient) FirstSpirit™ Version 5.0

<b>Version</b>	<b>1.09</b>
<b>State</b>	<b>RELEASED</b>
<b>Date</b>	<b>2013-09-04</b>
<b>Department</b>	FS-Core
<b>Copyright</b>	2013 <b>e-Spirit AG</b>
<b>File name</b>	USER50EN_FirstSpirit_UserDocumentation

e-Spirit AG  
Barcelonaweg 14  
44269 Dortmund | Germany

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

[info@e-spirit.com](mailto:info@e-spirit.com)  
[www.e-spirit.com](http://www.e-spirit.com)

**e-Spirit**

## Table of content

<b>1</b>	<b>Introduction.....</b>	<b>16</b>
1.1	Subject of this documentation .....	16
1.2	The FirstSpirit concept.....	19
1.2.1	Stores .....	19
1.2.2	Permission assignment .....	20
1.2.3	Multilingualism .....	21
1.2.4	Parallel access in multi-user environments.....	23
1.2.5	Versioning, historizing and archiving .....	24
<b>2</b>	<b>FirstSpirit start page (Java Web Start).....</b>	<b>25</b>
2.1	Login page.....	25
2.2	Start page .....	27
2.2.1	Quick Start area.....	27
2.2.2	Clients area.....	28
2.2.3	Administration area .....	28
2.2.4	User area.....	29
2.3	Project selection .....	33
<b>3</b>	<b>The FirstSpirit JavaClient user interface .....</b>	<b>35</b>
3.1	FirstSpirit menu bar .....	36
3.1.1	Project .....	37
3.1.2	Tasks .....	40
3.1.3	Search .....	41
3.1.4	CorporateContent .....	54



3.1.5	View.....	55
3.1.6	Extras.....	65
3.1.7	Help.....	75
3.2	The JavaClient horizontal tool bar .....	76
3.2.1	Back/Next.....	76
3.2.2	New bookmark.....	76
3.2.3	Home .....	78
3.2.4	Refresh.....	78
3.2.5	Editing mode on/off .....	78
3.2.6	Save .....	80
3.2.7	Preview .....	81
3.2.8	New .....	82
3.2.9	Delete .....	82
3.2.10	TabView .....	88
3.2.11	Show task list .....	88
3.2.12	Online Help .....	88
3.3	The vertical tool bar .....	89
3.3.1	The global search ("Search") .....	90
3.3.2	The tree structure ("Explore").....	99
3.3.3	The user-defined view ("Organize") .....	101
3.3.4	The Multisite Management area.....	112
3.4	FirstSpirit AppCenter/Integrated preview .....	112
3.4.1	Integrated preview for content .....	113
3.4.2	Integrated preview for media.....	116
3.4.3	FirstSpirit AppCenter.....	118
3.5	The editing area.....	120
3.5.1	Editing in several workspaces (horizontal tab navigation) .....	120



3.5.2	Breadcrumb navigation .....	122
3.5.3	Enhanced workspace-view ("TabView") .....	123
3.6	The project entry page .....	124
3.7	The status bar .....	124
3.8	Error display .....	125
<b>4</b>	<b>JavaClient Page Store.....</b>	<b>128</b>
4.1	General context menus in the Page Store .....	128
4.1.1	New .....	129
4.1.2	Lock/Unlock (Editing mode On/Off).....	132
4.1.3	Reset Changes .....	132
4.1.4	Cut .....	132
4.1.5	Copy .....	133
4.1.6	Paste .....	133
4.1.7	Rename.....	133
4.1.8	Delete .....	133
4.1.9	Display in current/new workspace .....	134
4.1.10	Version history .....	134
4.1.11	Workflow .....	134
4.1.12	Execute Script.....	134
4.2	Special context menus in the Page Store.....	134
4.2.1	CorporateContent .....	134
4.2.2	Add to Content transport feature (at folder and page level).....	135
4.2.3	Display preview (at page level).....	135
4.2.4	Show preview errors (at page level).....	135
4.2.5	Preview (release) (at page level) .....	135
4.2.6	Preview errors (release) (at page level).....	136



4.2.7	Export (at folder and page level) .....	136
4.2.8	Import (at root, folder and page level) .....	136
4.2.9	Restore deleted objects (at root, page, content area and folder level).....	136
4.2.10	Change position (at section level).....	137
4.3	Functions under the Extras context menu of the Page Store .....	138
4.3.1	Release (at root, folder and page level).....	138
4.3.2	Delete metadata (at root, folder and page level).....	138
4.3.3	Change permissions (at root, folder and page level) .....	138
4.3.4	Reset write protection.....	138
4.3.5	View Template (at page and section level).....	138
4.3.6	Select another template (at page and section level).....	139
4.3.7	Jump to source section (at section reference level).....	139
4.3.8	Edit validity period (at section level).....	139
4.3.9	Show usages (at page level).....	141
4.3.10	Change reference name (at root, folder and page level).....	141
4.3.11	Display properties .....	142
4.3.12	Cancel editing (at page and section level) .....	144
4.3.13	Display dependencies (at page and section level) .....	144
4.4	Functions under the Plug-ins context menu of the Page Store.....	145
4.4.1	Open FormView AppTab (at page and section level).....	145
4.5	Settings at page level.....	146
4.6	Settings at section level.....	148
4.6.1	Section references.....	149
<b>5</b>	<b>Content Store of the JavaClient.....</b>	<b>150</b>
5.1	General context menus in the Content Store .....	150



5.1.1	New .....	150
5.1.2	Editing on/off.....	153
5.1.3	Cut.....	153
5.1.4	Copy.....	153
5.1.5	Paste.....	154
5.1.6	Rename.....	154
5.1.7	Delete .....	154
5.1.8	Display in current/new workspace .....	154
5.2	Special context menus in the Content Store.....	155
5.2.1	Restore deleted objects .....	155
5.2.2	CorporateContent (at folder and data source level) .....	157
5.2.3	Add to Content transport feature .....	158
5.2.4	Add all displayed datasets to feature (data source) .....	158
5.2.5	Version history .....	158
5.2.6	Starting workflow .....	158
5.2.7	Execute Script.....	158
5.2.8	Show usages (at dataset level).....	159
5.2.9	New bookmark (at dataset level).....	159
5.2.10	Display dependencies (at dataset level) .....	159
5.3	Functions under the Extras context menu of the Content Store .....	159
5.3.1	Change permissions .....	159
5.3.2	Release displayed lines (data source) .....	159
5.3.3	View Template (data source) .....	160
5.3.4	Set filter (data source) .....	160
5.3.5	Delete data source .....	161
5.3.6	Reset write protection.....	162
5.3.7	Change reference name (data source) .....	162



5.3.8	Display dependencies (data source).....	163
5.3.9	Display properties .....	163
5.4	Functions under the Plug-ins context menu in the Content Store....	163
5.4.1	Open FormView AppTab (dataset).....	163
5.5	Tables in the Content Store .....	163
5.5.1	Filtered data sources .....	163
5.5.2	Data overview – Current data records.....	164
5.5.3	Dataset selection.....	168
5.5.4	Data entry .....	172
5.5.5	Data overview – Released records .....	175
<b>6</b>	<b>Media Store of the JavaClient.....</b>	<b>177</b>
6.1	General Media Store context menus.....	177
6.1.1	New .....	177
6.1.2	Editing on/off.....	181
6.1.3	Reset Changes .....	181
6.1.4	Cut.....	181
6.1.5	Copy.....	181
6.1.6	Paste.....	181
6.1.7	Rename.....	181
6.1.8	Delete .....	182
6.1.9	Display in current/new workspace .....	182
6.1.10	Version history .....	182
6.1.11	Workflow.....	182
6.1.12	Execute Script.....	183
6.2	Special Media Store context menus in the Media Store .....	183
6.2.1	Add to Content transport feature .....	183



6.2.2	Export (at folder and media level) .....	183
6.2.3	Import (at root and folder level) .....	183
6.2.4	Restore deleted objects (at root and folder level) .....	184
6.3	Functions under the Extras context menu of the Media Store .....	185
6.3.1	Release .....	185
6.3.2	Delete metadata .....	185
6.3.3	Change permissions .....	185
6.3.4	Reset write protection .....	185
6.3.5	Show usages (at media level) .....	185
6.3.6	Display properties .....	186
6.3.7	Make medium language-dependent/language-independent (at media level).....	186
6.3.8	Cancel editing .....	186
6.3.9	Change reference name .....	186
6.3.10	Reset stored URLs .....	187
6.3.11	Display dependencies (at media level) .....	188
6.4	Editing area at folder level .....	188
6.4.1	Overview tab.....	188
6.4.2	Properties tab.....	189
6.4.3	Messages tab.....	190
6.5	Editing area for images.....	190
6.5.1	List tab .....	192
6.5.2	Preview tab .....	192
6.6	Edit images.....	193
6.6.1	Functions in the editing area .....	194
6.6.2	Integrated cropping function .....	197
6.6.3	Processing of media in the Integrated preview .....	201





6.6.4	Java Image Editor .....	202
6.6.5	Simple image processing (PicMonkey) .....	209
6.6.6	Enhanced image processing (Pixlr).....	209
6.7	Editing area for files.....	211
6.7.1	Microsoft Office (Windows only).....	213
6.7.2	OpenOffice (BETA, not MacOS) .....	214
6.8	Media Import Wizard .....	215
6.8.1	Media Import Wizard – Step 1 of 9.....	215
6.8.2	Media Import Wizard – Step 2 of 9.....	216
6.8.3	Media Import Wizard – Step 3 of 9.....	217
6.8.4	Media Import Wizard – Step 4 of 9.....	217
6.8.5	Media Import Wizard – Step 5 of 9.....	218
6.8.6	Media Import Wizard – Step 6 of 9.....	219
6.8.7	Media Import Wizard – Step 7 of 9.....	219
6.8.8	Media Import Wizard – Step 8 of 9.....	220
6.8.9	Media Import Wizard – Preview (Objects tab) .....	223
6.8.10	Media Import Wizard – Preview (Import Properties tab).....	225
6.8.11	Media Import Wizard – Step 9 of 9.....	226
<b>7</b>	<b>Site Store of the JavaClient.....</b>	<b>227</b>
7.1	General Site Store context menus.....	227
7.1.1	New .....	227
7.1.2	Edit on/off .....	229
7.1.3	Undo.....	229
7.1.4	Cut.....	230
7.1.5	Copy.....	230
7.1.6	Paste.....	230



7.1.7	Rename.....	230
7.1.8	Delete .....	230
7.1.9	Display in current/new workspace .....	230
7.1.10	Version history .....	231
7.1.11	Workflow.....	231
7.1.12	Execute Script.....	231
7.2	Special context menus in the Site Store .....	231
7.2.1	Add to Content transport feature.....	231
7.2.2	Set as default start menu (folder level only) .....	231
7.2.3	Set as Start page (page reference level only).....	232
7.2.4	Reference to another page (page reference level only).....	232
7.2.5	Preview (page reference and document group level) .....	232
7.2.6	Show errors (page reference and document group level) .....	232
7.2.7	Preview (release) (page reference and document group level).....	233
7.2.8	Preview errors (release) (page reference and document group- level).....	233
7.2.9	Export (at folder and page reference level) .....	233
7.2.10	Import (at root and folder level) .....	233
7.2.11	Restore deleted objects (at root and folder level) .....	233
7.3	Functions under the Extras context menu of the Site Store.....	235
7.3.1	Release.....	235
7.3.2	Delete metadata .....	235
7.3.3	Change permissions .....	235
7.3.4	Reset write protection.....	236
7.3.5	View page (page reference level only) .....	236
7.3.6	Show usages (at folder and page reference level) .....	236
7.3.7	Display properties .....	236



7.3.8	Edit page (page reference level only) .....	236
7.3.9	Cancel editing .....	236
7.3.10	Change reference name .....	237
7.3.11	Reset stored URLs (at menu and page reference levels) .....	238
7.3.12	Display dependencies .....	238
7.4	Settings at menu level .....	238
7.4.1	Names tab .....	238
7.4.2	Menu order tab .....	242
7.4.3	Page groups tab .....	242
7.4.4	Messages tab .....	243
7.4.5	Variables tab .....	244
7.5	Settings at page reference level .....	246
7.5.1	Languages tab .....	246
7.5.2	Content tab .....	251
7.6	Document group settings .....	253
7.6.1	General details .....	255
7.6.2	Contents .....	255
7.6.3	Template settings .....	256
7.6.4	Presentation channels .....	256
<b>8</b>	<b>Template Store of the JavaClient .....</b>	<b>257</b>
<b>9</b>	<b>Global Settings .....</b>	<b>259</b>
9.1	Global content area .....	259
9.2	Project settings .....	260
9.3	URL settings .....	260
9.3.1	"SEO URLs" tab .....	262



9.3.2	"Short URLs" tab .....	267
9.4	User settings .....	271
9.4.1	Browser tab .....	271
9.4.2	Editors tab .....	272
<b>10</b>	<b>The default input forms .....</b>	<b>273</b>
10.1	Single-line text .....	274
10.2	Multi-line text .....	274
10.3	Rich text editor .....	275
10.4	Tables .....	277
10.5	Numbers .....	280
10.6	Date/time selection .....	280
10.7	Reference selection .....	282
10.8	Link input .....	285
10.8.1	Internal Link .....	286
10.8.2	External link .....	288
10.8.3	Content Link .....	289
10.8.4	Link to a related project .....	290
10.9	List creation .....	291
10.10	Dataset selection .....	294
10.11	Checkbox .....	295
10.12	Selection list .....	296
10.13	Radio button .....	296
10.14	Toggle .....	297
10.15	Multiple selection list .....	298
10.16	Button .....	299
10.17	Link-sensitive graphic .....	300



10.18	Default values and invalid entries .....	302
10.18.1	Default values .....	302
10.18.2	Invalid entries .....	303
10.19	Enhanced functions in input forms.....	304
<b>11</b>	<b>General use .....</b>	<b>305</b>
11.1	Task list .....	305
11.2	The message board.....	307
11.3	Metadata .....	309
11.4	Drag-and-drop functions of FirstSpirit.....	310
11.4.1	Move using "drag-and-drop" .....	310
11.4.2	Copy using "drag-and-drop" .....	310
11.4.3	Create section references in the Page Store .....	311
11.4.4	Drag-and-drop from the local file system into JavaClient.....	311
11.4.5	Drag-and-drop from JavaClient into the local file system.....	312
11.4.6	Drag-and-drop within the thumbnail view (media) .....	313
11.4.7	Drag-and-drop between two workspaces.....	314
11.4.8	Inter-Store drag-and-drop.....	315
11.4.9	Drag-and-drop from the search dialog into JavaClient.....	316
11.4.10	Drop onto media input forms .....	316
11.4.11	Drop onto text input forms.....	317
11.4.12	Drag-and-drop within input forms.....	317
11.4.13	Drag-and-drop between two input forms .....	318
11.4.14	Drag-and-drop whole input forms .....	319
11.4.15	Drop onto elements which are not yet open in a workspace...	320
11.4.16	Drop onto elements which are not in edit mode .....	320
11.4.17	Drag-and-drop via the Content Store overview .....	320



11.4.18	Drag-and-drop from the integrated preview.....	321
11.4.19	Drag-and-drop from Microsoft applications.....	321
11.5	Working with the rich text editor input form.....	322
11.5.1	Icon bar in the rich text editor.....	322
11.5.2	Context menu in the rich text editor.....	322
11.5.3	Spell check in the rich text editor.....	324
11.5.4	Lists: Generating lists.....	325
11.5.5	Convert selected text into list.....	330
11.5.6	Exit/interrupt list.....	333
11.5.7	Delete/merge lists.....	336
11.5.8	Copy/Move lists.....	341
11.5.9	Insert links.....	342
11.5.10	Integrate tables in the rich text editor.....	343
11.6	Working with the table input form.....	348
11.6.1	Icon bar of a table.....	348
11.6.2	Context menus in tables.....	349
11.6.3	Checking spelling in tables.....	349
11.6.4	Cell properties.....	350
11.7	Selection dialogs.....	351
11.7.1	Navigation tab.....	351
11.7.2	Search tab.....	355
11.7.3	Navigation using the keyboard.....	359
11.7.4	Multiple selection in selection dialogs.....	360
11.8	Working with media galleries.....	362
11.9	Keyboard shortcuts.....	363
11.9.1	Keyboard shortcuts in the rich text editor.....	365
11.9.2	Keyboard shortcuts for DOM table.....	367



11.10	Version Management in the FirstSpirit JavaClient .....	369
11.10.1	Terms and concepts.....	369
11.10.2	Version history functions.....	371
11.10.3	Version history in the Page Store .....	380
11.10.4	Version history in the Content Store .....	382
11.10.5	Version history in the Media Store.....	383
11.10.6	Version history in the Site Store .....	384
11.11	Show dependencies via reference graph .....	386
<b>12</b>	<b>Workflows in the FirstSpirit JavaClient .....</b>	<b>390</b>
12.1	FirstSpirit standard workflows .....	390
12.2	Starting a workflow.....	391
12.2.1	Context-bound workflows .....	391
12.2.2	Workflows without context.....	393
12.3	Release of objects (standard sequence) .....	393
12.3.1	Request a release via the context menu .....	394
12.3.2	Request a release via the task list.....	398
12.4	Special release options .....	399
12.4.1	Example: Specific release - Ensure accessibility (parent chain) option .....	402
12.4.2	Example: Specific release – Release recursively .....	403
12.4.3	Example: Specific release – Release dependent objects .....	404
<b>13</b>	<b>Permissions in the FirstSpirit JavaClient .....</b>	<b>407</b>
13.1	Editorial permissions .....	407
13.1.1	Editorial permissions through group membership .....	409
13.1.2	Permission assignment in the FirstSpirit JavaClient .....	409



13.1.3	Inheritance of editorial permissions within the project.....	416
13.1.4	Possible editorial permissions.....	417
13.1.5	Dependences between editorial permissions.....	425
13.2	Permissions to execute workflows.....	427
13.2.1	Permission assignment in the FirstSpirit JavaClient.....	427
13.2.2	Define permissions for all workflows.....	428
13.2.3	Define permissions for individual workflows.....	429
13.2.4	Change authorized groups/users.....	431
13.2.5	Permission assignment for executing the transitions.....	432
13.3	User permissions.....	434
13.3.1	User permissions through group membership.....	434
13.3.2	Permission assignment in the FirstSpirit JavaClient.....	435
13.3.3	Evaluation of the user permissions.....	436
13.3.4	Inheritance of user permissions within the project.....	437
13.3.5	Dependencies of user and editorial permissions.....	440



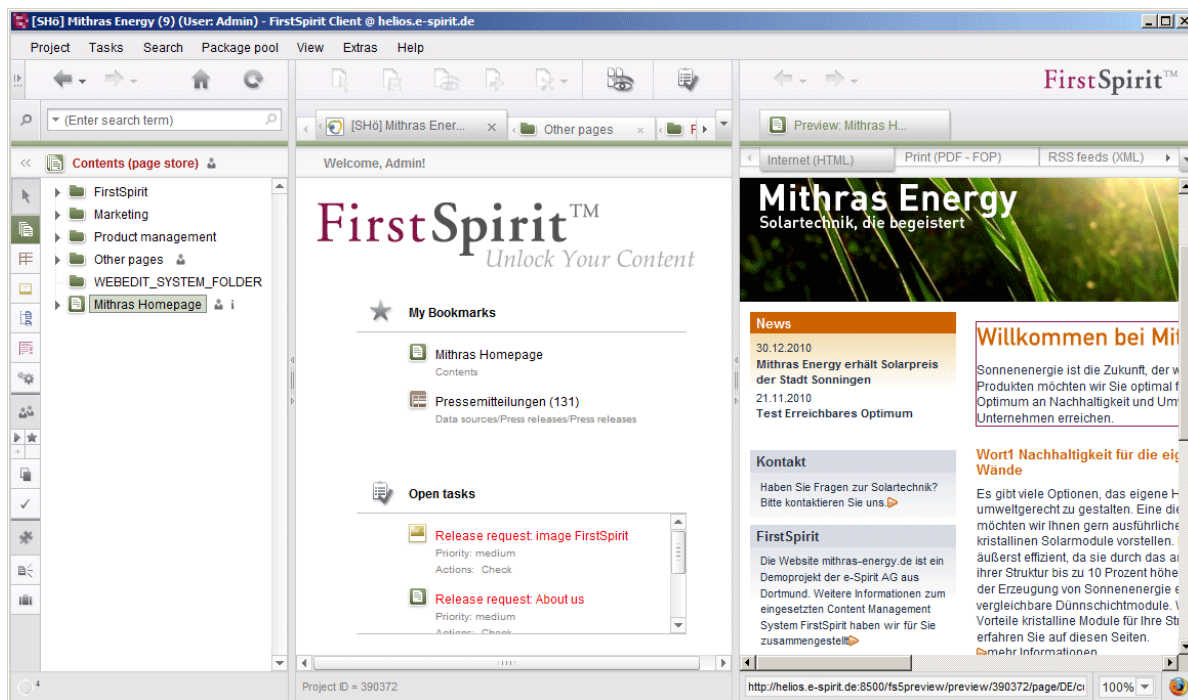


# 1 Introduction

## 1.1 Subject 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 three, the elements of the tree structure (in the left-hand column of the screen) are described first, followed by the editing screens for the individual objects and other functions in the center column and finally the functions in the right-hand column.



**Figure 1-1: FirstSpirit JavaClient**

The options available for starting JavaClient are explained in **section 2** (see section 2, page 25).

The individual elements of the JavaClient interface are described in **section 3** (see section 3, page 35).



**Section 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 setting options for pages and sections (see section 4, page 128).

**Section 5** deals with the Content Store which is used for acquiring, entering and managing highly structured content (see section 5, page 150).

**Section 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 section 6, page 177).

**Section 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 section 7, page 227)

**Section 8** is a general description of how the Template Store functions (see section 8, page 257).

In **section 9** the functions of the areas Global Content (maintenance of small components of pages), Project Settings (definition of replacement rules), URL Settings (definition of URLs for pages) and User Settings (integrate Editors and Browsers) are described (see section 9, page 259).

**Section 10** provides a detailed overview of the standard FirstSpirit input components (see section 10, page 273).

**Section 11** deals with the functions for general use of FirstSpirit (see section 11 page 305).

The function of workflows is described in **section 12** (see section 12, page 390)

In **section 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 section 13, page 407).





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



*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 implemented. 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.*



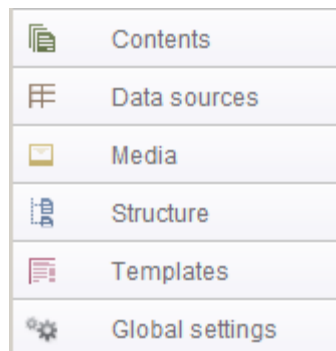
## 1.2 The FirstSpirit concept

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 images, files, structured data, layout and navigation structure) is managed in separate areas (stores). This concept fulfills 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 any knowledge of HTML and XML.

FirstSpirit JavaClient is divided into six stores each with different colors, which are strictly separated from each other but are functionally dependent:



**Figure 1-2: Stores in FirstSpirit JavaClient**

- The Page Store contains all pages and their editorial content (see section 4, page 128).
- The Content Store is used to create highly structured pages by managing the content using database mechanisms (see section 5, page 150).
- The Media Store contains all media files used in the project. These do not necessarily have to be conventional 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 section 6, page 177).

- The Site Store determines the navigation structure of the website (see section 7, page 227).
- The Template Store includes all information concerning the layout and functions of the website (see section 8, page 257).
- The Global Settings contain global user and project settings, global content (with multiple use) and definitions of URLs (see section 9, page 259).

## 1.2.2 Permission assignment

In addition, FirstSpirit provides a clear 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 section 13.1.2, page 409).

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 section 13.1 page 407).
- **Permissions to execute workflows:** These are a special type of editorial permissions which refer to the workflows within a project only (see section 13.2 page 427).
- **User permissions:** These are permissions valid for the "user" of the site generated with FirstSpirit. User permissions are always linked with the personalization system used (see section 13.3, page 434).



### 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:

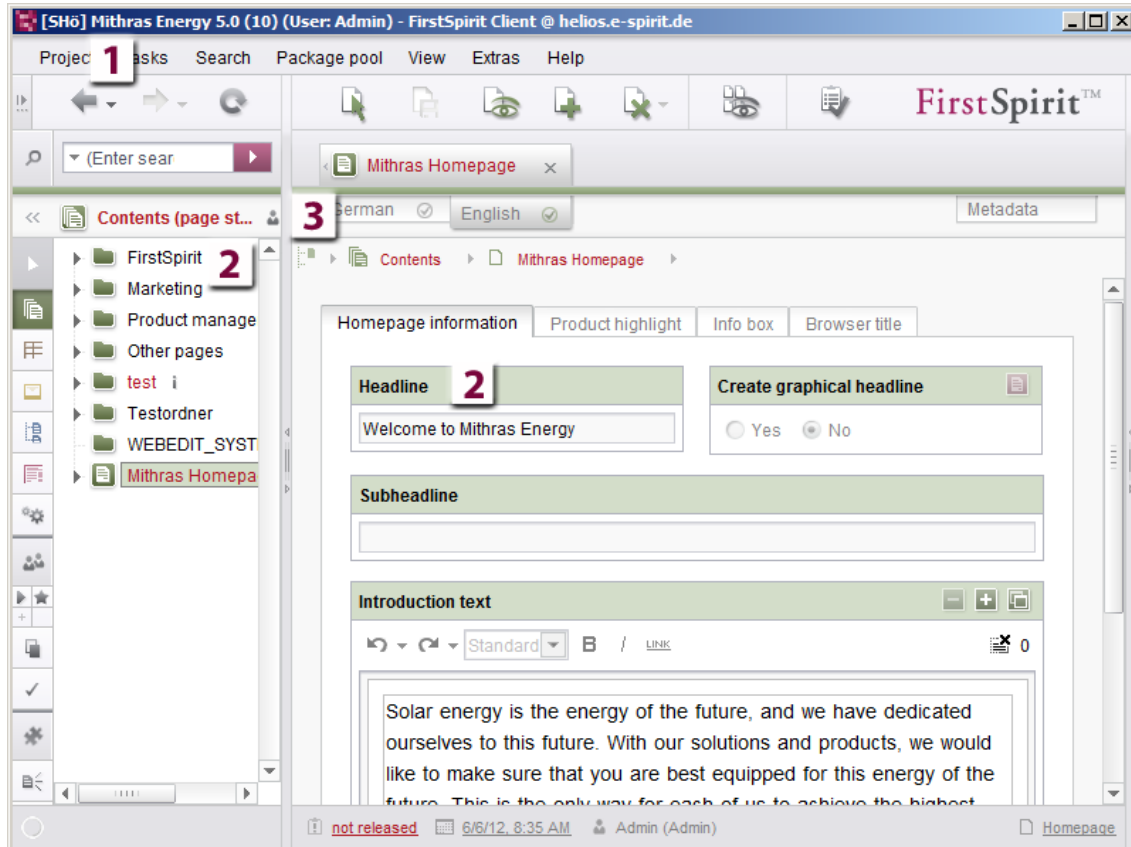


Figure 1-3: Language setting in JavaClient

1. **Language setting of FirstSpirit JavaClient ("Locale"):** This setting is defined using the combo box of the FirstSpirit Start page (see section 2.2 page 27) 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 labeling 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. You can currently choose from Dutch, English, French, German, Italian, Spanish and Russian.
2. **Editing languages:** Editing languages are defined for a project by the project administrator and can then be configured by the editor using the "View – Preferred



Display Language" menu (see section 3.1.5.2 page 55). 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 (labeling of the input fields, tool tips, elements of a combo box, 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 (per tab), 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 images (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 simultaneously for an object:

- Create: Create new objects in the project.
- Delete: Delete existing objects from the project.
- Copy: Copy existing objects to another position within the project.
- Move: Move 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 (see also section Release of objects (standard sequence), page 393).

The "Editing mode" system implemented 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 Editing 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 interlinked in the Site Store to form a 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 01.01.2012 by (temporarily) setting the current live project to the status of 01.01.2012 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 is frequently focused on collating content in self-contained units and subsequent transfer to a long-term storage medium.



## 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 login 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 section 2.2.4.1 from page 30. Further information on the configuration options is given in the FirstSpirit Manual for Administrators, section 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.



You have logged off successfully. Please log on again.

Please use the "Automatic logon" button to log on automatically without entering your user name and password.

Automatic Login

User

Password


Login

FirstSpirit™ is a product of the e-Spirit AG. Legal notices

Server name: FirstSpirit 5.0 (QS) - Server-Version: 5.0

**Figure 2-1: Java Web Start – Login**

The bottom 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. Dutch, English, French, German, Italian, Spanish and Russian are available here.

Clicking on **Automatic Login** automatically logs the user in to the FirstSpirit Server under the user's Windows login.

**User:** In this field, enter the user name assigned to the user on the FirstSpirit Server.

**Password:** Enter the password for the user 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 user name that was entered.



## 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 section 2.2.1).
- Clients area (see section 2.2.2).
- Administration area (see section 2.2.3).
- Users area (see section 2.2.4).



**Figure 2-2: Java Web Start – Start page**

The bottom 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. Dutch, English, French, German, Italian, Spanish and Russian are available here.

### 2.2.1 Quick Start area

The left-hand part of the page contains the quick-start entries that 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 Clients area

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

- **JavaClient (editing system):** Click this entry to start the FirstSpirit editing system. A project selection dialog appears, from which the editor can select the desired project. This project is then opened in the JavaClient.

When first starting JavaClient the following confirmation dialog can be displayed:



Activate the option "Always trust content from this publisher" to prevent that this dialog will not be shown again.

- **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 entries for administrators are located at the top right of the window.

- **Server monitoring:** Click this entry to open FirstSpirit Server Monitoring, which monitors the FirstSpirit Server. A detailed description of this is given in the "FirstSpirit Server



Monitoring" section of the *Documentation for Administrators*.

- **Server and project configuration:** Click this entry to open the FirstSpirit Server and Project Configuration, which helps FirstSpirit administrators with general, administrative tasks relating to FirstSpirit. A detailed description of this is given in the "FirstSpirit Server and Project Configuration" section of the *Documentation for Administrators*.



*This area is hidden from users who do not have administrator rights.*

## 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 user who is currently logged in:

- **Connection settings:** The connection settings of the user who is currently logged in can be changed here (see section 2.2.4.1 page 30).
- **Change password:** The password of the user who is currently logged in can be changed here (see section 2.2.4.2 page 32).
- **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 section 2.2.4.3 page 33).
- **Log out:** Click this entry to quit the current FirstSpirit session for the user who is logged in (see section 2.2.4.4 page 33).
- **Help:** Click this entry to open the FirstSpirit online documentation.



## 2.2.4.1 Configuring connection settings

FirstSpirit™  
Unlock Your Content

English

Use settings

Mode: HTTP

Use HTTPS protocol:

Host:

Port:

Memory: 512

Compression: 0 (none)

Encryption: 0 (none)

Servlet zone:

Optional parameters:

Save Cancel

FirstSpirit™ is a product of the e-Spirit AG. Legal notices

Server name: FirstSpirit 5.0 (QS) - Server-Version: 5.0\_DEV.18.50003 - User: shoebbel

Figure 2-3: Configuring connection settings

To change the settings on this page, first activate the **Use settings** checkbox.



*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 user who is currently logged in via the pull-down list:

- HTTP: normal Internet connection (default setting)
- Socket: direct connection mode.

**Use HTTPS protocol:**

**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. You can choose 512, 768, 1024 or 1536 MB.



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

- 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 user who is currently logged in:

- None: No encryption for the transfer of data between client and server.
- TLS<sup>1</sup>: 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 user who is currently logged in. 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

---

<sup>1</sup> 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.



*This entry is only available to users who were created manually on the server, not to external users who were created by an automatic SSO login.*

Figure 2-4 shows a dialog box for changing a password. The dialog has a title bar and a close button. It contains the following elements:

- A text label "Password" followed by an empty input field.
- A red error message: "Please enter a new password for user 'Admin':".
- A text label "New password" followed by an empty input field.
- A text label "Enter the password again" followed by an empty input field.
- Two buttons: "Change" and "Cancel".
- A red error message at the bottom: "Please enter the current password!".

**Figure 2-4: 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.*

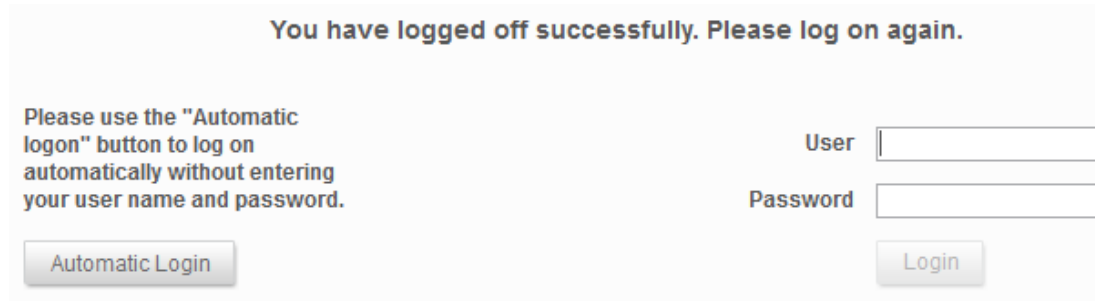
**Enter the password again:** The new password is entered again in this field to exclude accidental typing errors when changing the password.

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



### 2.2.4.3 Change user

Click this entry to log in another user to the server. The login page opens again (see section 2.1 page 25) which is now extended to include the automatic login option.



The screenshot shows a login interface with the following elements:

- Message: "You have logged off successfully. Please log on again."
- Text: "Please use the "Automatic logon" button to log on automatically without entering your user name and password."
- Buttons: "Automatic Login" and "Login"
- Form fields: "User" and "Password" input boxes.

**Figure 2-5: 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 the **Automatic Login** button to log in under the Windows login.

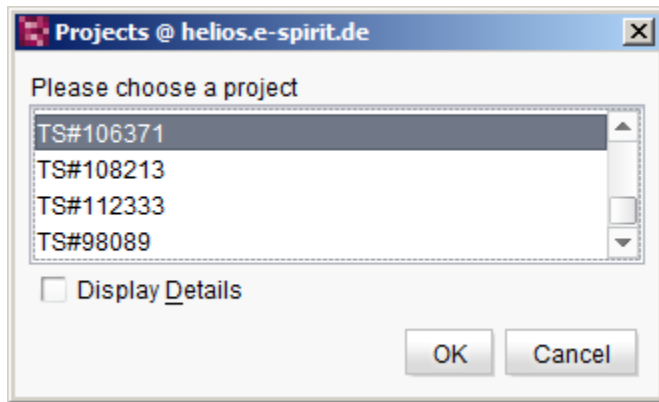
### 2.2.4.4 Log out

By clicking this entry the logged in user can log out from the server. The login page then opens again (see section 2.1, page 25).

## 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 "OK" button. With the project selection the specific data and settings of the project are then loaded.

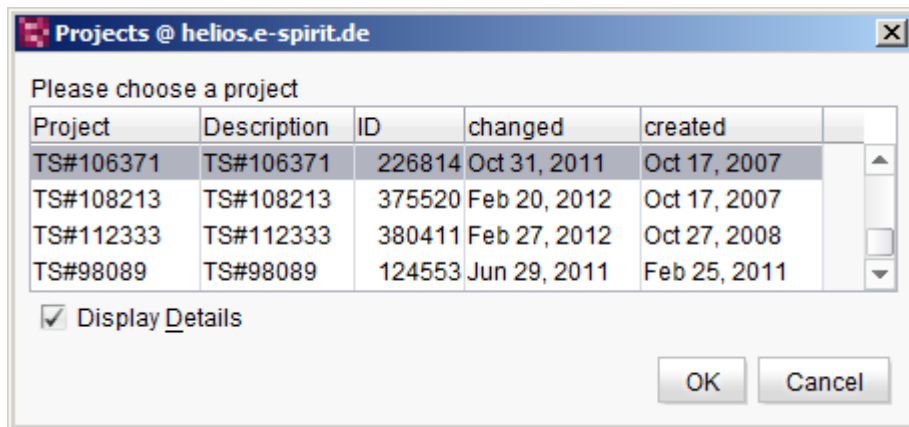




**Figure 2-6: Project selection**

You can start the desired project by clicking OK or double-clicking the relevant entry.

If the "Display details" checkbox is activated a further dialog box appears with further information on the displayed projects.



**Figure 2-7: Project details**

Confirming a selection of a selected table cell with the "OK" button opens the corresponding project. If the "Display details" checkbox is deactivated the project selection dialog from Figure 2-6 appears again.

The start dialog that now appears 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 FirstSpirit version and the server name.



### 3 The FirstSpirit JavaClient user interface

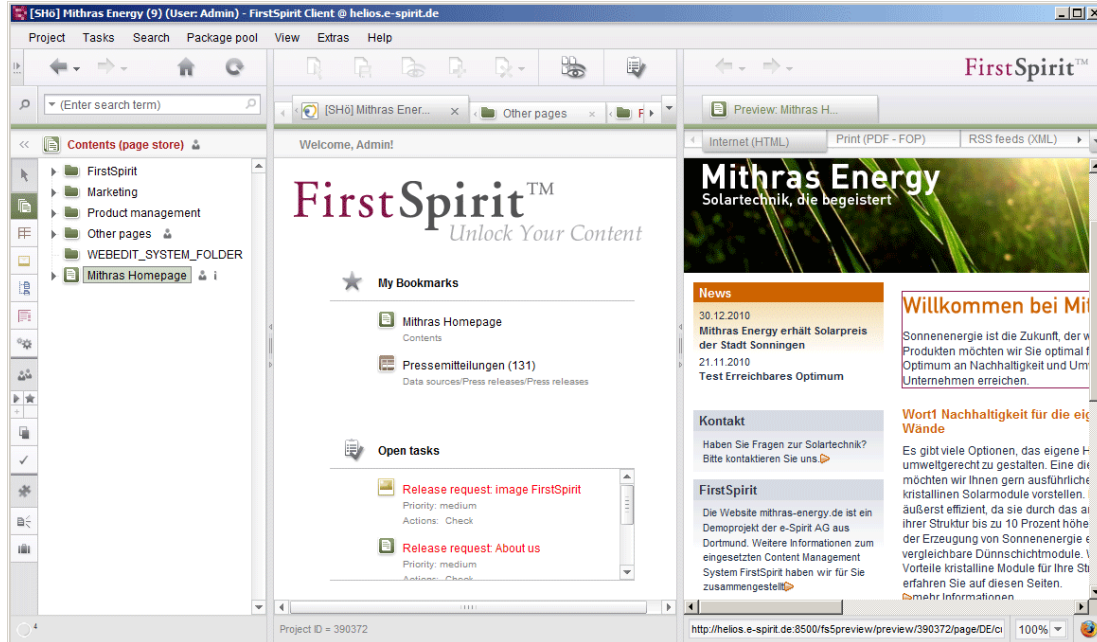


Figure 3-1: Project view in the FirstSpirit JavaClient

Apart from the FirstSpirit logo, the JavaClient **title bar** 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 **menu bar**. Explanations on the menu bar are given in section 3.1 page 36. Directly below the menu bar is the **horizontal tool bar** which provides several frequently required functions as icons. The tool bar functions are described in section 3.2 page 76.

The FirstSpirit JavaClient consists of a screen view divided vertically into three sections of varying sizes relative to one other:

Various project contents can be displayed in the left-hand screen area, e.g. the **tree structure**, **search results**, **bookmarks**, **tasks**, the **clipboard** etc. When a project starts, the project's tree structure is displayed in the special Page Store. The display in the left-hand column is controlled via the vertical tool bar on the left-hand edge of the screen. This bar is described in detail in section 3.3, page 89. The global search is located above the project content display. This is described in section 3.3.1, page 88. The project contents that can be displayed in the left-hand screen column are described in section 3.3.1, page 88.

The center of the screen is the **editing area**, which directly refers to the activated position within

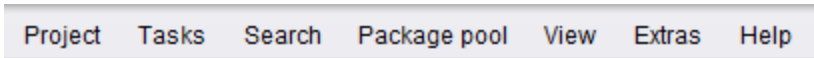


this tree structure. This contains the editing screens for the individual objects from the tree structure. When a project is opened for the first time, the editing area contains a personalized project entry page (see section 3.6, page 124) with an overview of the user's bookmarks, open tasks and actions.

The right-hand area of the screen contains the **FirstSpirit AppCenter**. Depending on the project settings, you can examine FirstSpirit content here in an integrated browser (**integrated preview**) in order to check how the content would be displayed on the website. You can also access third-party applications here, such as Microsoft Office. See also section 3.4, page 112.

The **status bar** is located in the bottom part of the FirstSpirit JavaClient. This is where information about the object that is currently selected in the tree structure is displayed (see section 3.7, page 124). This is also where errors are displayed (section 3.8, page 125).

### 3.1 FirstSpirit menu bar



**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 sections with a brief description and any limitations.

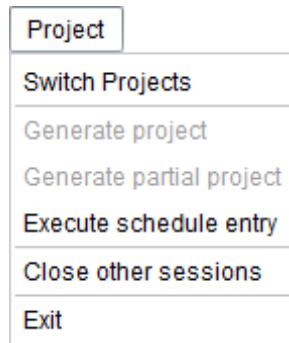


*If the functions of individual menu items are not available they are displayed in "gray". Possible reasons for this include: activated or deactivated Editing mode, the status of the current object and the permissions assigned to the user.*

The most important and most commonly used functions in the menu bar are also included in the tool bar as buttons. The tool bar is located directly below the menu bar and is described in section 3.2 page 76.



### 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 (section 2.3, page 33). A new project can then be selected and opened in the dialog box.



*A warning is issued if projects in the open project are still in editing 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 display "locked objects" (section 3.1.3.2, page 43).*

#### 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.*

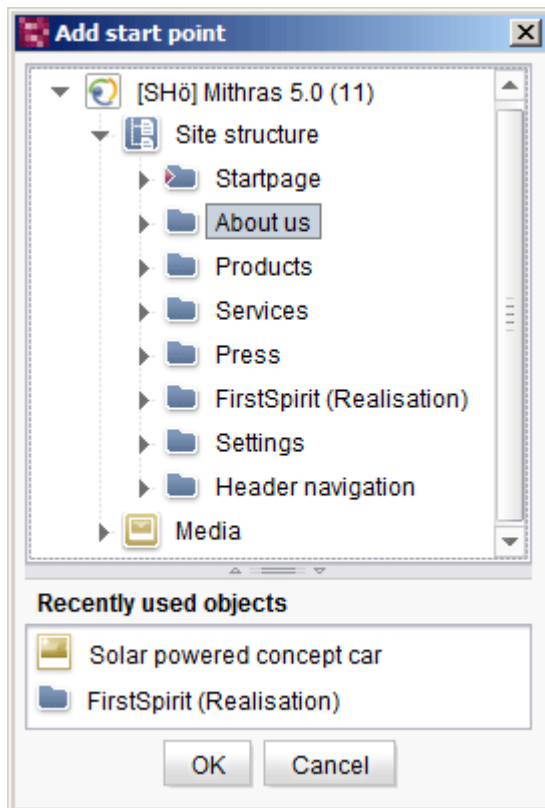
#### 3.1.1.3 Generate particular project

From time you may only want to generate a subset of a certain project. This type of partial generation can be started via this menu entry. Which nodes are taken into account depends on the configuration in the project properties (see *FirstSpirit Manual for Administrators*, "Perform



Generation" section).

If the project administrator has set an option for the user in the project properties, they can select (in a new window) the pages of the Site Store or nodes from the Media Store to be generated. By selecting a node from the Media Store, 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 triangle symbols 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.

Access to generated project content is restricted in FirstSpirit. When the URL is opened in the browser, it is therefore possible to first display a login dialog. If the editor is already logged in to the system, the authentication information is automatically transferred (opening in the browser by clicking on the URL button, for example).





In addition to the nodes selected in the **Add start 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 generation on the FirstSpirit Server can be deleted completely.
- The generation can be canceled if no changes have been made to the selected node since the last generation or if generation is prevented by media.
- Only files that 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 an option for the project, the generation schedule starts as soon as the **Generate partial 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 Execute schedule entry

This menu entry can be used to schedules directly (on schedules, see also *FirstSpirit Manual for Administrators*, "Schedule management" section). Clicking this menu entry opens a dialog box with a list of all schedules for this project that can be started directly.



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

#### 3.1.1.5 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.







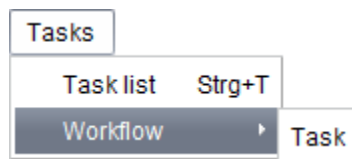
*Changes that have not been saved are lost when a session is ended.*

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

### 3.1.1.6 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 that objects are still in editing 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 section 11.1, page 305.

The task list can also be opened using the  icon in the tool bar or with Ctrl + T.

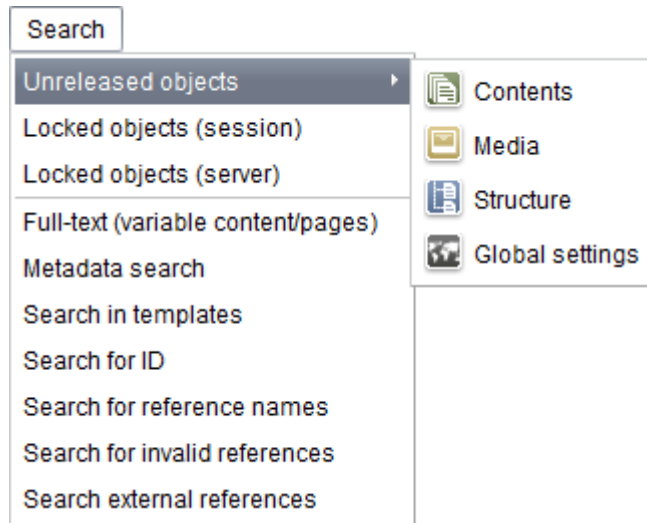
### 3.1.2.2 Workflow

This menu function can be used to start what are known as workflows without context. Workflows without context are workflows that 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 section 12.2.2 page 393.



### 3.1.3 Search

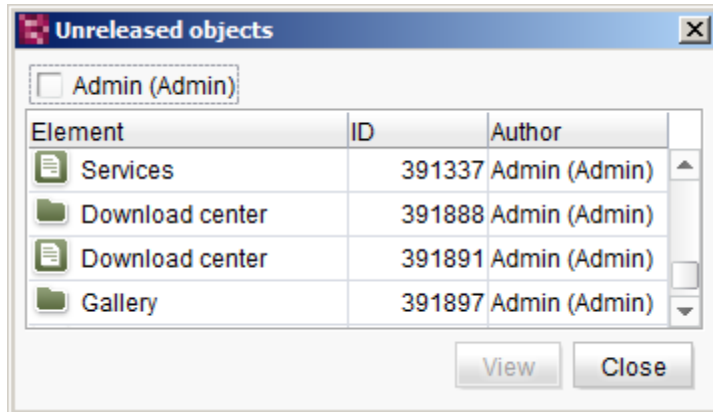


**Figure 3-6: Main menu item – Search**

- Unreleased objects see section 3.1.3.1
- Locked objects (session) see section 3.1.3.2
- Locked objects (server) see section 3.1.3.3
- Full-text (variable content/pages) see section 3.1.3.4
- Metadata search see section 3.1.3.5
- Search in templates see section 3.1.3.6
- Search for ID see section 3.1.3.7
- Search for reference name see section 3.1.3.8
- Search for invalid references see section 3.1.3.9
- Search for external references see section 3.1.3.10



### 3.1.3.1 Unreleased objects



**Figure 3-7: List of 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 (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.

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



## 3.1.3.2 Locked objects (session)

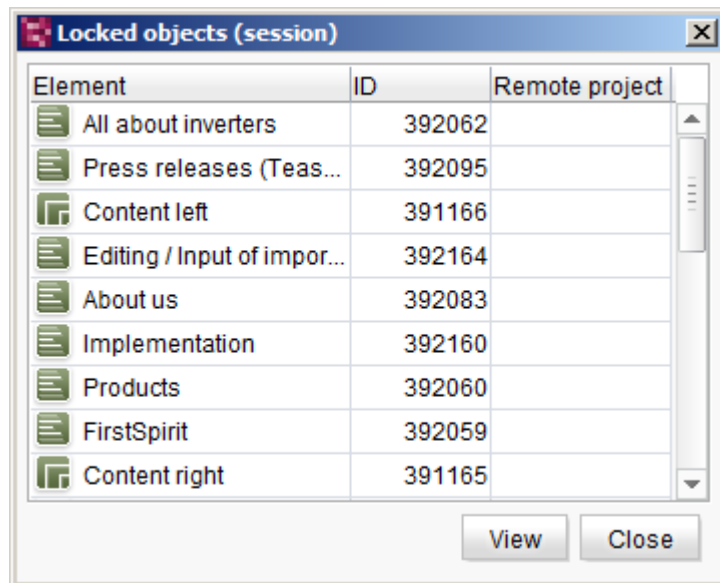


Figure 3-8: List of all locked objects in a session

The user can click this menu item to display all the objects of a project for which they themselves have activated Editing mode.

If an object in the selection list is selected, click the **View** 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 (server)

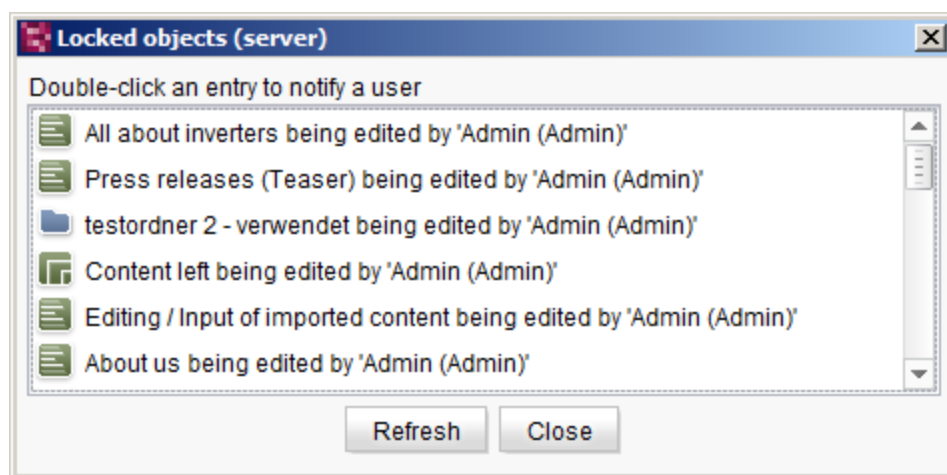
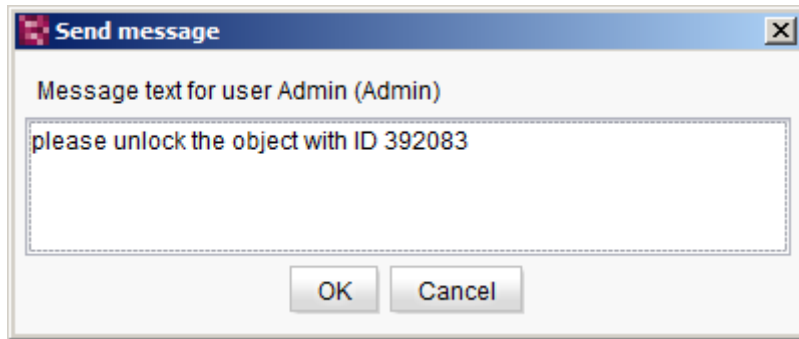


Figure 3-9: List of all locked objects



By clicking this menu item, the user can display all objects in a project which are currently in editing 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 box.



**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. The text can be edited before being sent, if necessary.

#### 3.1.3.4 Full-text (variable content/pages)

This function can be used to search all of the project's stores for specific content. 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:

- Full-text search (see section 3.1.3.4.1 page 45)
- Editor search (see section 3.1.3.4.2 page 46)
- Result (see section 3.1.3.4.3 page 47)



*Not all document types are supported in a Media Store search as a matter of principle and for technical reasons.*

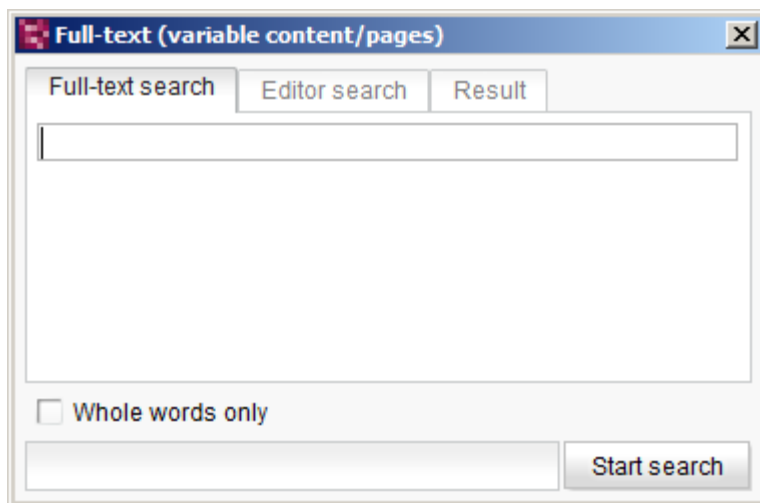


**Search with wildcards:** One or several parts of the words in the search term can be replaced with the \* wildcard.

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

### 3.1.3.4.1 Full-text search

The "Full-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:

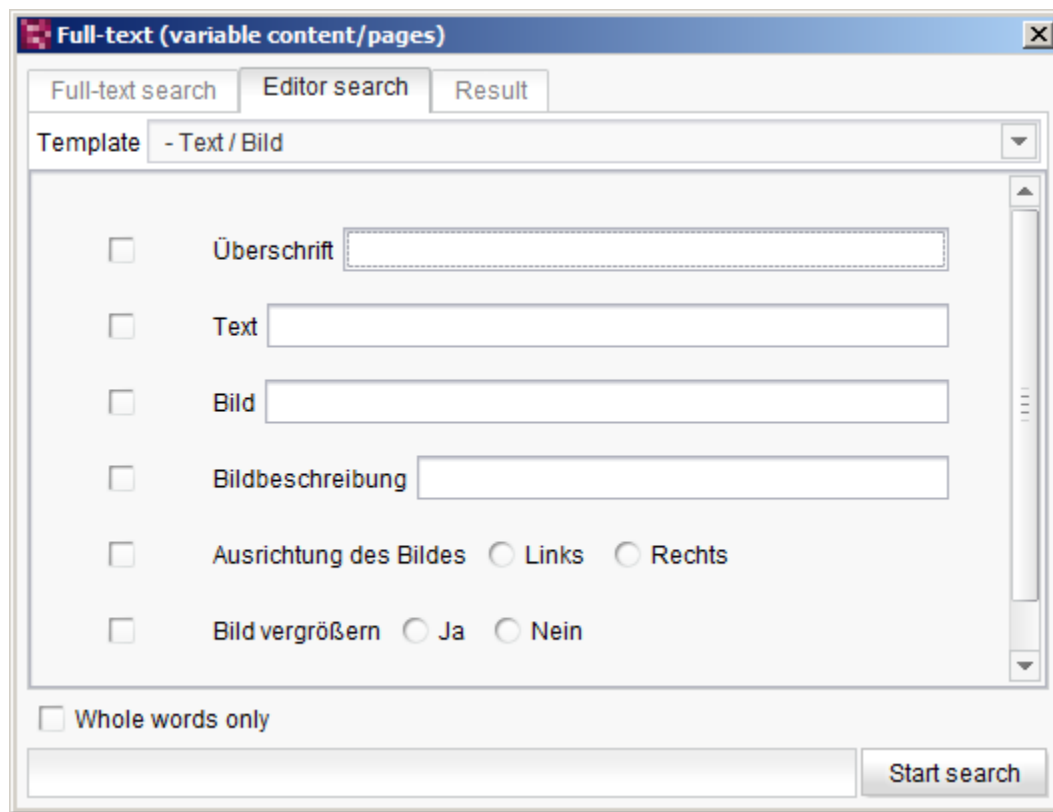
Search text	Result (activated)	Result (deactivated)
Motor	Motor	Motor, Motorway
Motorway	Motorway	Motor, Motorway



**Start/stop 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.

In the **Image selection** input component it is possible to enter the text that is searched for in the component's reference line. For example, if the word "media" is searched for without the option "Whole words only", all metadata elements in which an image 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.

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:

- Full-text search (see section 3.1.3.5.1 page 48)
- Editor search (see section 3.1.3.5.2 page 48)
- Result (see section 3.1.3.5.3 page 48)

**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 Full-text search

The "Full-text search" tab can be used to search for text fragments located in any object within the metadata.

Full-text search in the metadata is analogous to the full-text search in the Full-text search (see section 3.1.3.4.1 page 45).

### 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.

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 section 3.1.3.4.2 page 46).

### 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 section 3.1.3.4.3 page 47).



### 3.1.3.6 Search in templates

This function can be used to search the Template Store of the project for specific content.

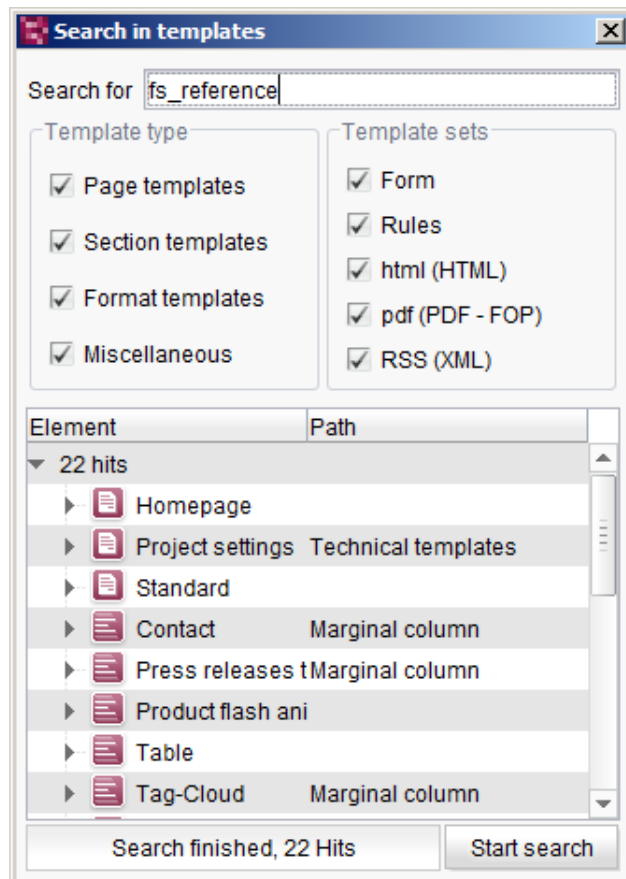


Figure 3-13: Search for Templates

**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 check mark 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 check mark 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.

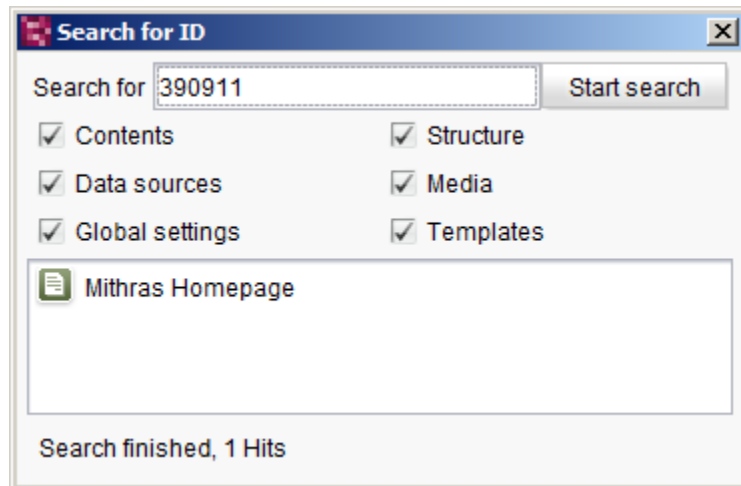
Within the search results the templates found can be expanded and contracted analogous to the tree display. Double-click a search result to switch the focus directly to the corresponding element in the tree view. The tab in which the search term was found is displayed in the client's editing area.

**Start/stop 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 3-14: 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 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 check mark in front of several stores is deactivated these stores are excluded from the search. The search function will therefore display results quicker.

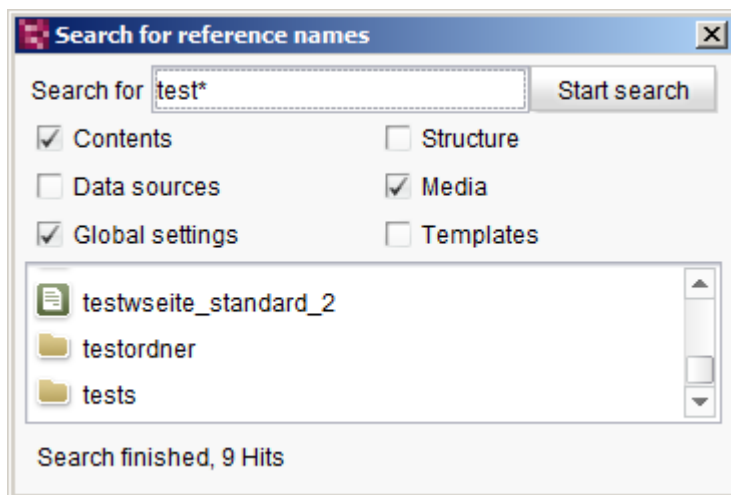


3.1.3.8 Search for reference name

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 "View – Preferred display language" menu (see section 3.1.5.2 page 55).*

*Reference names can also be accessed by the keyboard shortcut **ALT + P** (see section 11.9, page 363).*



**Figure 3-15: Search for reference name**

The search for reference names is performed in the same way the search for ID (see section 3.1.3.6 page 49).

**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

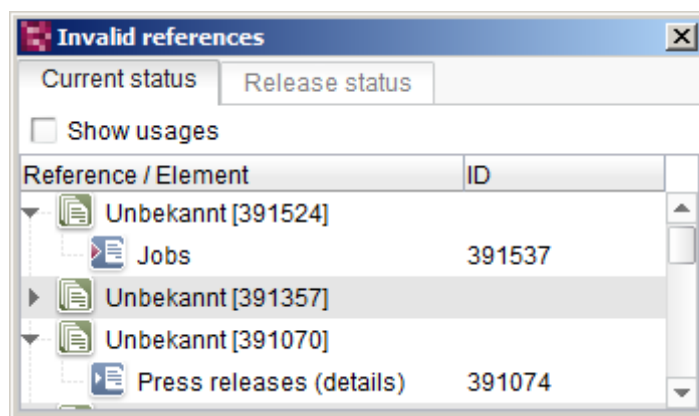


M\*way\*

Motorway, Motorway junction

### 3.1.3.9 Search for invalid references

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



**Figure 3-16: Search for invalid references**

All invalid references in the project are displayed in the overview. According to the project status they are divided into invalid references in the current (i.e. not released) project status and invalid 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 invalid 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 invalid 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:*



## 3.1.3.10 Search for external references

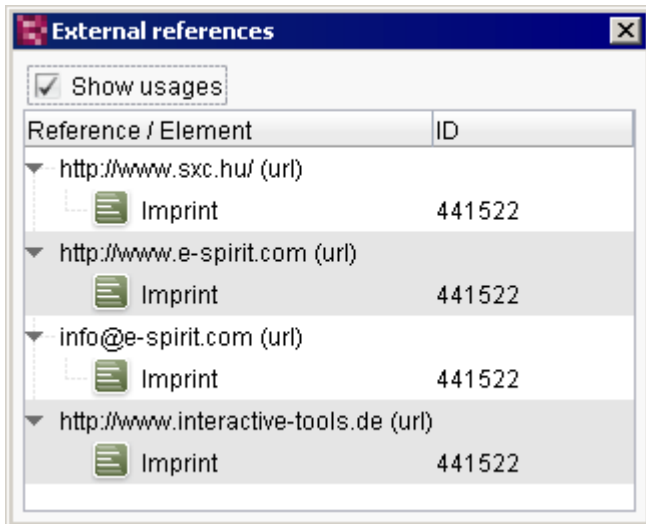


Figure 3-17: Search for external references

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 data source and the ID of the dataset are displayed. It is possible to tell which type of reference it is from the icons. Depending on which link template was used for the external link, the suffix "url" shows that the link in question is an Internet address and "email" shows that it is an e-mail address.

**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 CorporateContent

The function "FirstSpirit CorporateContent" is a license-dependent FirstSpirit function.

It makes it possible to group together objects from the FirstSpirit Client (such as pages, including all their links) into what are known as packages and to prepare them for importing to various target projects.

The separate *FirstSpirit CorporateContent* module documentation contains detailed descriptions of FirstSpirit CorporateContent.



### 3.1.5 View

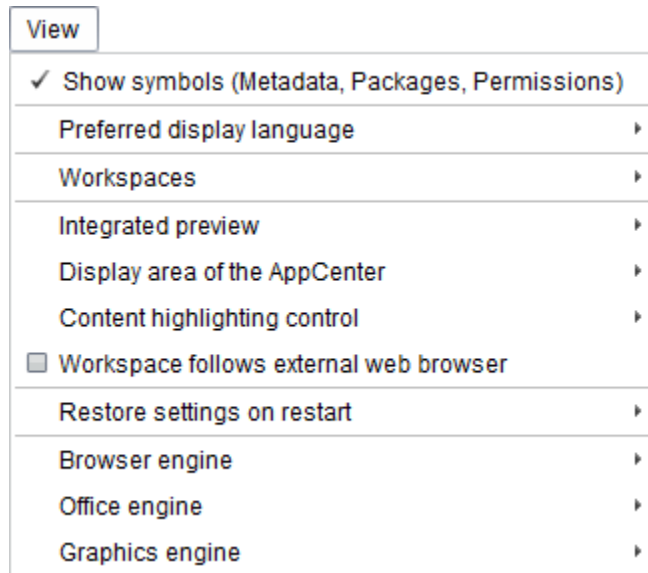





Figure 3-18: Main menu item – View

#### 3.1.5.1 Show symbols (Metadata, Packages, Permissions)

If this menu function is activated, all nodes to which permissions have been assigned are marked by the icon . In addition, all nodes for which metadata is defined are denoted by the icon . Objects managed with the FirstSpirit CorporateContent (formerly "Package Management") are denoted by the icon  (further information of FirstSpirit CorporateContent is given in the *FirstSpirit CorporateContent* documentation).

#### 3.1.5.2 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 toolbar or dialogs etc. are determined by the language settings on the Start page (locale language) (see section 2.2 page 27).*

**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 labeling of the Page Store's input components (e.g. text input fields). If this menu function is deactivated, the input components are labeled in the respective project language (content language). In this case, the labeling 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.

*For information on multilingualism in FirstSpirit projects, see also section 1.2.3, page 21.*



*Each FirstSpirit object has a reference name, which must be unique in each store.*

*When creating new objects in JavaClient reference names are made automatically unique by default by adding a number, e.g. "\_1".*

*When creating new objects in WebClient reference names of pages (Page Store) and page references (Site Store) will be made unique by attaching an underscore and a random twelve-digit string, e.g. "\_xitefo542bdt".*



*Reference names can also be called on the selected object using the keyboard shortcut **ALT + P**.*

### 3.1.5.3 Visible project languages

This menu item is only visible if the administrator has allowed to hide project languages in the project configuration. Use this menu item to define which project languages are to be visible in



the JavaClient.

If the visibility is deactivated for a language the respective language tab in single stores will then no longer be displayed and the content can no more be edited in this language. However, the content of hidden project language will be shown in the preview.



### 3.1.5.4 Workspaces

Working with workspaces (see section 3.5.1 page 120) 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.

**Synchronize new with tree:** If this option is activated, the active workspace in the editing area of the FirstSpirit JavaClient is automatically synchronized with the tree view. Switching tabs in the editing area therefore has a direct effect on the tree navigation.

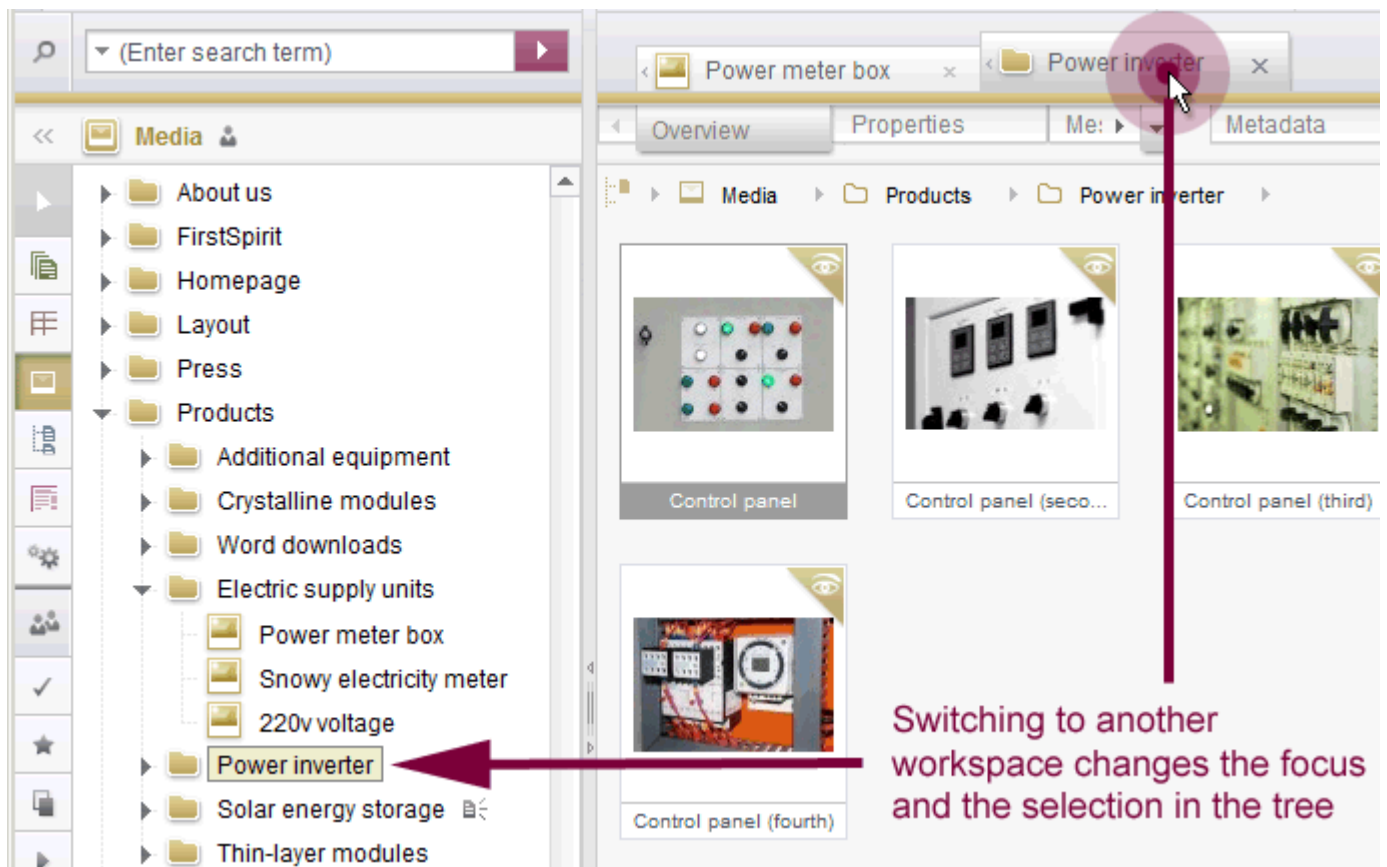


Figure 3-19: Tree synchronization on changing a workspace

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

This option can also be set for single tabs (see section 3.5.1 page 120).



**Save upon 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.

#### 3.1.5.5 Integrated preview

Using this menu item the integrated preview (see section 3.4 page 112) can be activated. If the option is activated the integrated preview can be configured using the menu item "Display area of the AppCenter" (see section 3.1.5.6 page 59).

If the **use for content** option is selected, the integrated preview for content from the Page Store and the Site Store is used. If this option is deactivated, the "Content highlighting control" (see section 3.1.5.7 page 60) and "Browser engine" (see section 3.1.5.10 page 63) 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 section 3.4.2, page 116). In this case, the menu item "Office engine" (see section 3.1.5.11 page 63) becomes active.



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

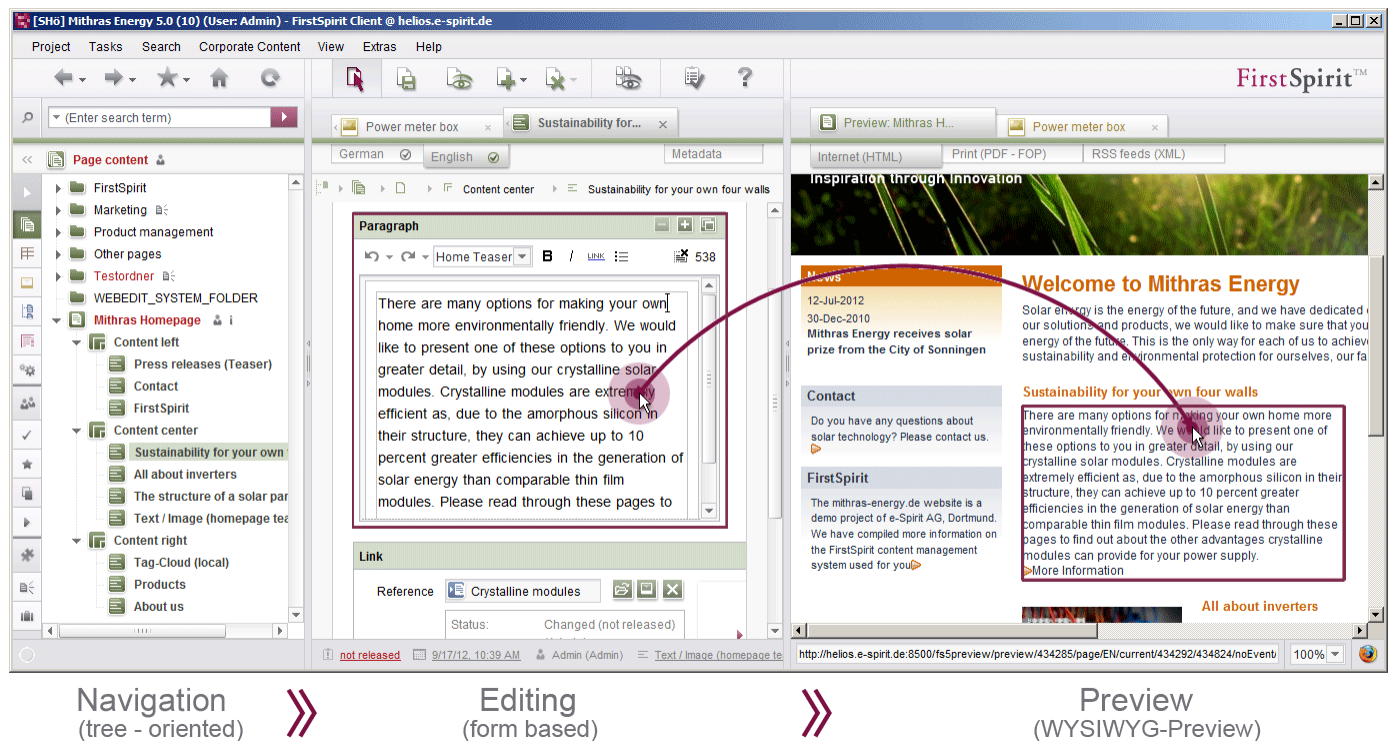
#### 3.1.5.6 Display area of the AppCenter

If the integrated preview is activated (see section 3.1.5.5, page 59) it can be optionally displayed on the right next to the workspace (**in this window** option) or, on smaller monitors, in a separate window (**in separate window** option). (Note: The "Content Highlighting" function is not supported if the integrated preview is displayed in an external window – see section 3.1.5.7.)



### 3.1.5.7 Content highlighting control

"Content Highlighting" provides orientation to the editor in the project: If, for example, the editor is in the form-based workspace of the JavaClient, for example within an input component, the currently edited area is highlighted in color 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 images can be quickly and easily found and changed.



**Figure 3-20: Content Highlighting**

The highlighted information depends on the context in which it is opened. Apart from individual input components it is possible, for example, to highlight the complete page information in the integrated preview by clicking a page. On the other hand, if a content area or a section is selected, only the content of the content area or 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 image 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 (see section 3.1.5.7, page 60), 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 component (click)
- Edit component (shift + click)
- **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":

- **Workspace ↔ preview:** Workspace follows preview and vice versa, i.e. clicking the active workspace displays the matching element in the integrated preview and vice versa.
- **Workspace → preview:** Preview follows workspace, i.e. clicking in the active workspace displays the matching element in the integrated preview, but not vice versa.
- **Workspace ← preview:** Workspace follows preview, i.e. clicking in the integrated preview



displays the matching element in the active workspace, but not vice versa.

- **Disable:** Content Highlighting is 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 permanently preselected.*



*To use the "Content Highlighting" functionality, the template developer first has to adjust a project's templates. The functionality only affects the preview of the HTML presentation channel. Other presentation channels are not taken into account.*



*Content Highlighting is not supported for display of the integrated preview in an external window (see section 3.4, page 112).*

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

#### 3.1.5.8 Workspace follows external web browser

If this menu item is activated and the integrated preview (see section 3.1.5.5 page 59) 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. Changes to the content of a page can therefore be made quickly and easily.

#### 3.1.5.9 Restore settings on restart

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 upon exiting") (see section 3.1.5.4, page 58). 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.*

#### 3.1.5.10 Browser engine

If the "Integrated preview – use for content" option is enabled (section 3.1.5.5, page 59), you can use this menu item to choose if Mozilla Firefox or Microsoft Internet Explorer is to be used for the integrated preview on Windows platforms. Microsoft Internet Explorer provides the same functions as Mozilla Firefox (see section 3.4 page 112). The selected browser is indicated in the bottom right-hand corner of the integrated preview.

If Mozilla Firefox is used, a special version of Firefox that is integrated in the JavaClient is used, so it is not necessary for Mozilla Firefox to be installed locally on the workstation. However, if Microsoft Internet Explorer is used, the browser must be installed on the workstation. The user-specific configuration is also used here. Check the *Technical Data Sheet* to see which versions are currently supported.

#### 3.1.5.11 Office engine

If the "Integrated preview - use for media" option is enabled (section 3.1.5.5, page 59), this menu can be used to set which application is to be used for Microsoft Office and OpenOffice file formats. In order to use Microsoft Office and OpenOffice, these applications must be installed on the workstation.

- **Microsoft Office (Windows only):** If this option is enabled, the relevant Microsoft Office application is used to display and 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 section 6.7.1, page 213).
- **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 MacOS (see section 6.7.2, page 214).





- **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 previously.



*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, section 3 "The FirstSpirit AppCenter".)*

### 3.1.5.12 Graphic engine

If the option "Integrated preview - use for media" is enabled (section 3.1.5.5, page 59), you can decide via this menu, which application is to be used for which image file format.

- **Java Image Editor:** If this option is enabled, the Java Image Editor is used for displaying and editing images (see section 6.6.4, page 202).
- **Simple image processing (PicMonkey):** If this option is enabled, the online image processing service [www.picmonkey.com](http://www.picmonkey.com) is used for displaying and editing images. PicMonkey enables simple, intuitive alteration of images (see section 6.6.5, page 209).
- **Enhanced image processing (Pixlr):** If this option is enabled, the online image processing service [www.pixlr.com](http://www.pixlr.com) is used for displaying and editing images. Pixlr is modeled more professional image editing software like Adobe Photoshop (see section 6.6.6, page 209).
- **disabled:** If this option is enabled, the familiar image processing functions are available.



*Use of the **PicMonkey** and **Pixlr** editors requires an active Internet connection. Further information is given on the corresponding Internet sites.*



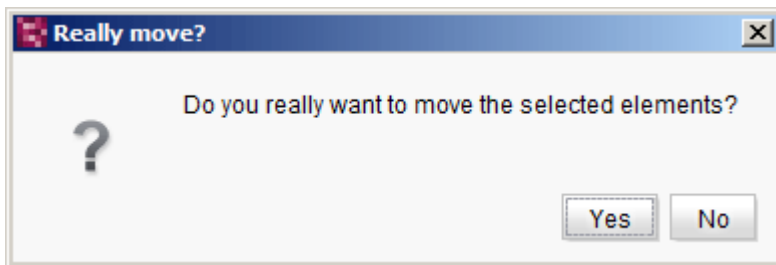


*If using applications in the Integrated preview, please note that FirstSpirit provides the interfaces required for the application integration, but in general does not have any influence on the integrated applications themselves. Integrated external applications are not part of the FirstSpirit product. 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, section 3 "The FirstSpirit AppCenter".)*

### 3.1.6 Extras

#### 3.1.6.1 Confirm Move operations

This menu function is activated by default. When objects are moved by drag-and-drop or cut and paste, the following question is displayed:



**Figure 3-21: Confirm Move operations**

This prevents multiple elements within the project from being moved inadvertently. 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 section 7.1.1 page 227), this confirmation dialog is not displayed.

#### 3.1.6.2 Show preview errors

If this menu function is activated, any errors are automatically displayed when a preview is rendered.



### 3.1.6.3 Show preview warnings

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

### 3.1.6.4 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 Content Store, as well as in the Global Settings.

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.

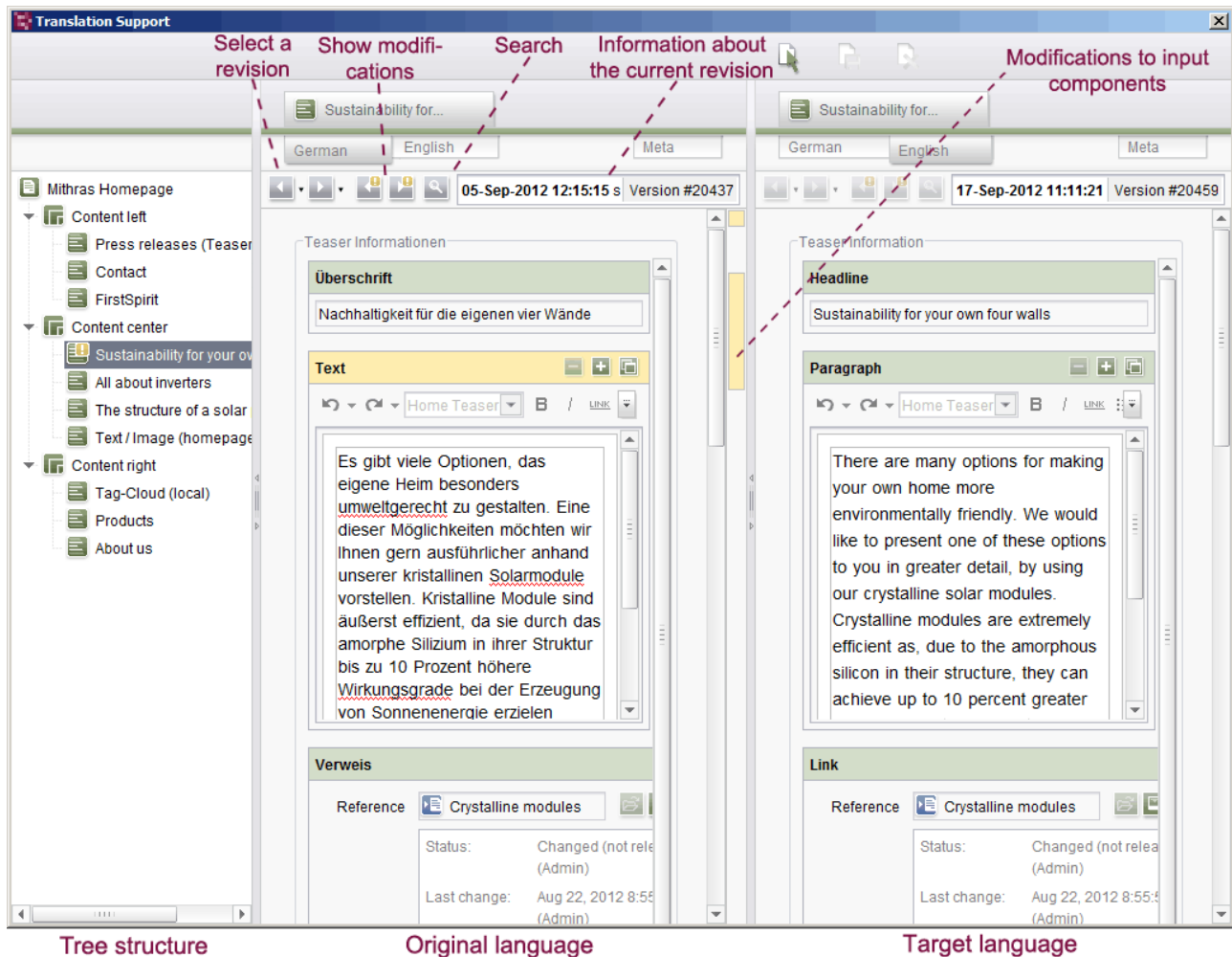


Figure 3-22: Translation help – Page Store




In the case of pages/sections, the tree structure in the **left-hand part of the window** is shown in a separate window area. The input components for the page and the relevant sections can be reached via this structure. The input components of the selected object are displayed directly in the View or Edit window.

Icons are used to visualize whether and what type of differences exist between the compared versions. For example, 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.

The **central part of the window** is the view area for a language A (source language). 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.

Within the input components, changes between different versions are visualized by a colored 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.

The **right-hand part of the window** is the view area for a language B (target language). 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.

With help of the  icon, content can be copied from the source 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.

Rules for the input of content which have been defined by the template developer (see Chapter 10.18.2 page 303) apply also in the translation help. Invalid entries are highlighted in color and they will be justified textually.

**Above the view area** for the source and target languages, the respective revision is displayed with the version number, date, time, last editor and the comment assigned to the revision ("Revision information"). The last version of the current object is displayed by default in each case. 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.

**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 tool tip shows the name of the relevant input component and it is possible to jump directly to the relevant input component with a click.

### 3.1.6.5 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.

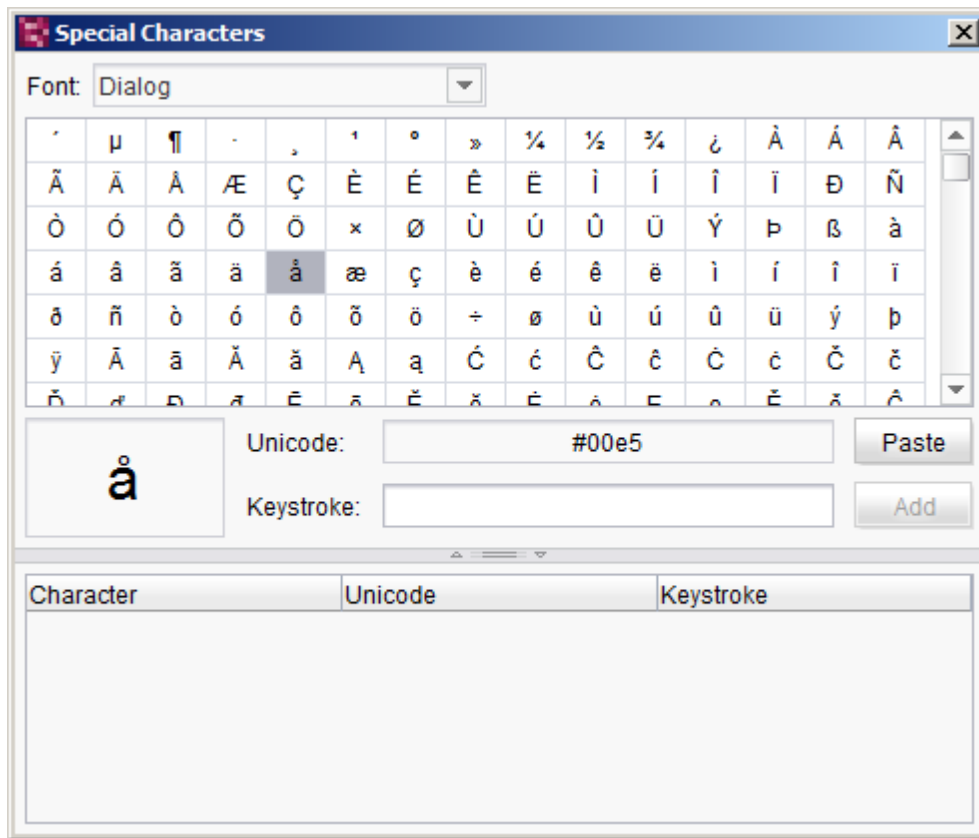


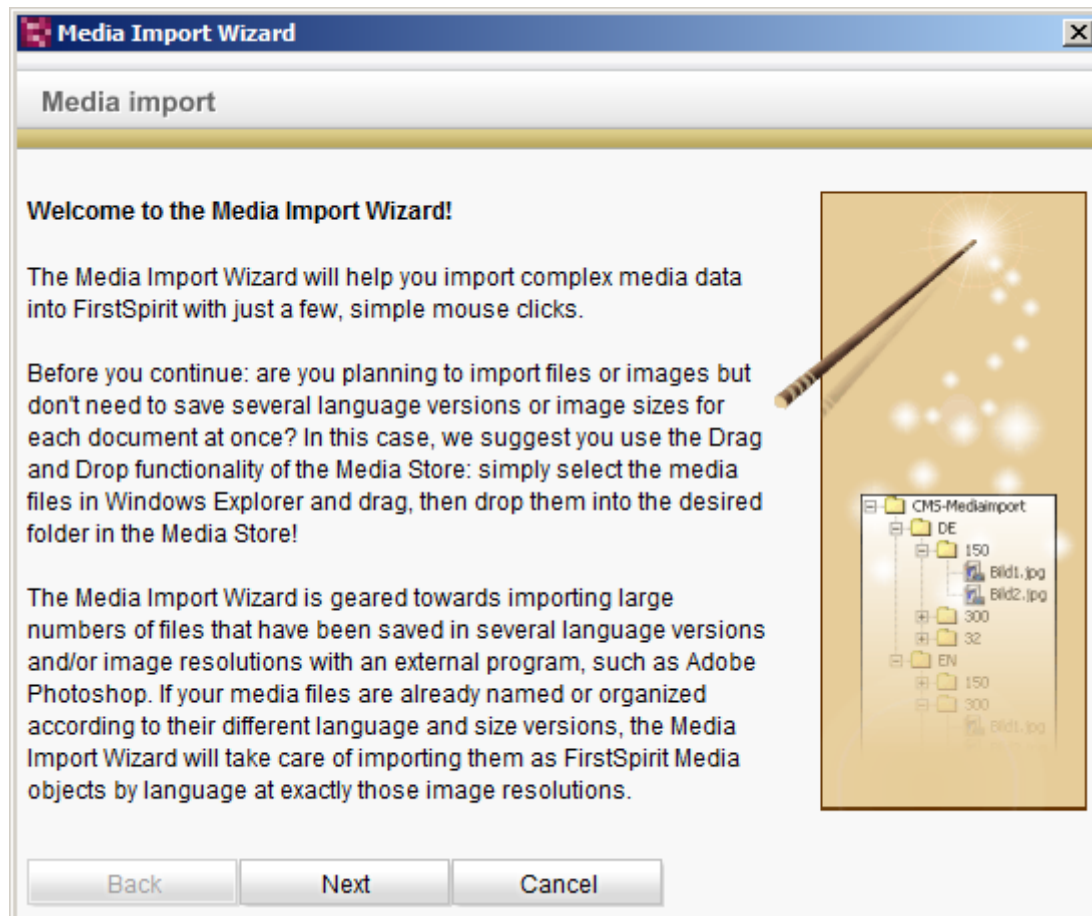
Figure 3-23: Inserting special characters

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.



### 3.1.6.6 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 3-24: 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 section 6.8, page 215.

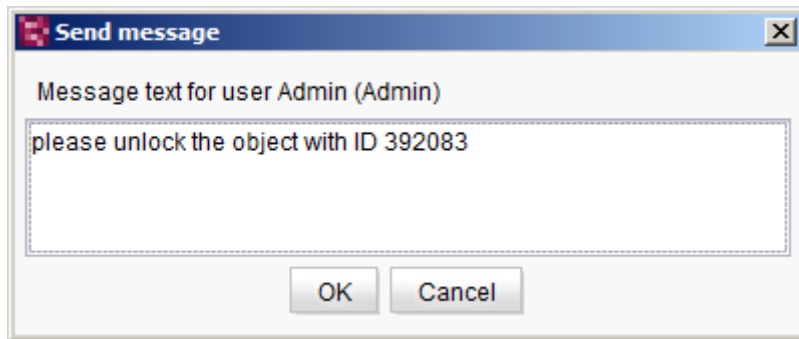




*If language-independent media only is to be imported with original resolution, it is possible to move the folder with the relevant media directly from the computer's file system into the Media Store using drag-and-drop. The files in the folder are then automatically imported into the Media Store folder. A differentiation is made between images and files and they are created accordingly. For information about the effects of media restrictions see section 6.1.1 page 177.*

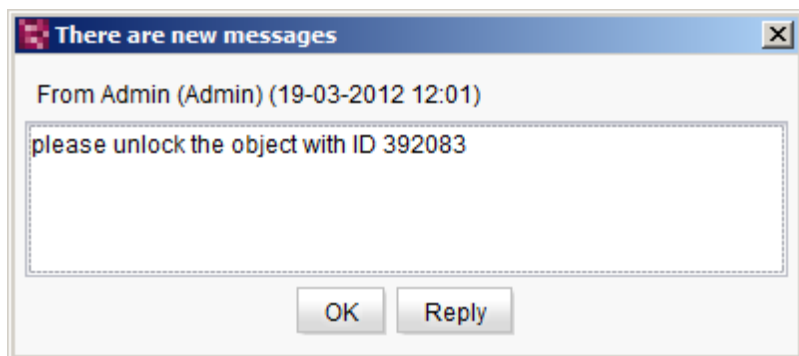
### 3.1.6.7 Send message

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 3-25: Send message**

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



**Figure 3-26: Message received**

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



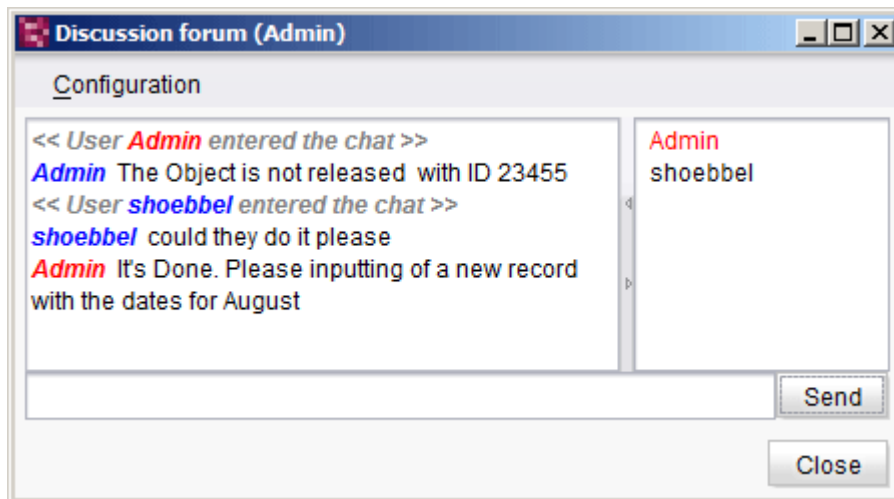
**Reply to all:** If the message was sent to several users simultaneously, this button can be used to send a message to all recipients (except yourself) of the original message.



*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.*

### 3.1.6.8 Discussion forum

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



**Figure 3-27: Discussion forum**

### 3.1.6.9 Copy FirstSpirit URL

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

### 3.1.6.10 Go to FirstSpirit URL

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.





### 3.1.6.11 Execute script

Scripts that are available to the user at this point are listed under this menu item. Scripts enable pre-programmed actions or calculations to be executed.

### 3.1.6.12 Extended logging

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

### 3.1.6.13 Print

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).



## Print preview window

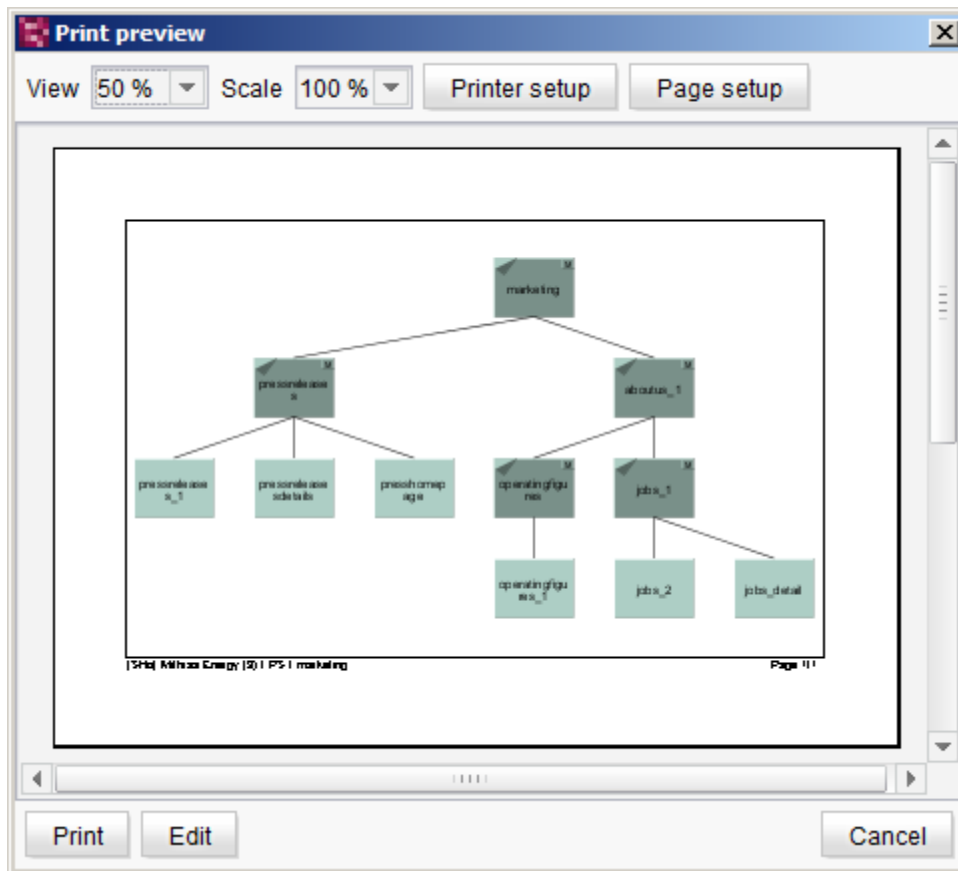


Figure 3-28: Print preview

**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.

**Scale:** 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

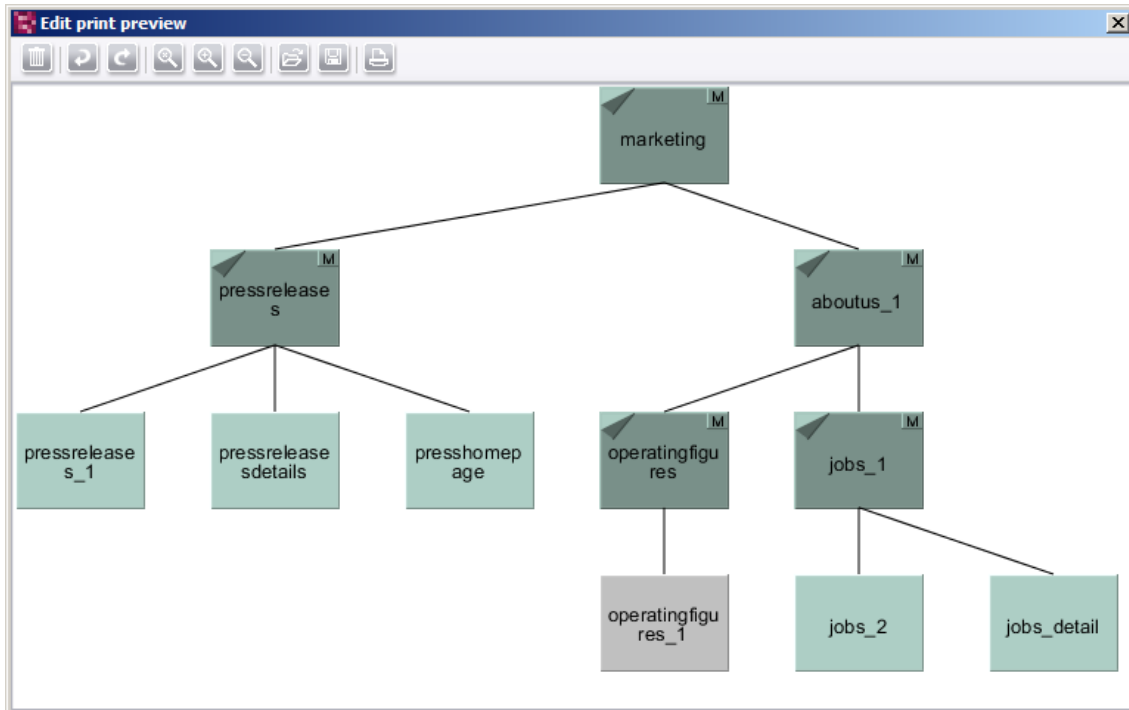
**Cancel:** Click this button to cancel printing

**Edit:** Click this button to open a further window in which the selected subtree can be limited or





specific elements can be highlighted:


### Edit Print Preview window





**Figure 3-29: Edit print preview**


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


 Undo (Ctrl + Z); this icon can be used to undo a range of changes made in the Print Preview.

 Restore (Ctrl + Shift + Z); 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 (Ctrl + L); 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 (Ctrl + S); this icon can be used to save the current print preview in the computer's file system.



Print preview (Ctrl + P); use this icon to return to the Print Preview window.

### Context menu in the Edit Print Preview window

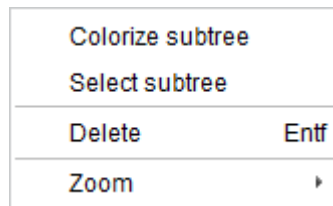


Figure 3-30: Context menu in the "Edit print preview" window

**Colorize subtree:** After activating this function a color scheme appears; all selected elements are marked with the color selected here.

**Select subtree:** This function can be used to select a tree element including all the elements below it to e.g. move this whole subtree to another position in the print view.

**Delete (Del):** 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.

### 3.1.7 Help

This menu entry can be used to access various different entry points of the FirstSpirit help documentation, information about the FirstSpirit version and environmental parameters used and a list of available keyboard shortcuts.



## 3.2 The JavaClient horizontal tool bar



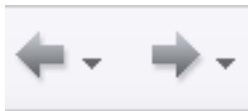
**Figure 3-31: Horizontal tool bar of the FirstSpirit JavaClient**

The horizontal tool bar 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 tool bar and are shown in gray.*

### 3.2.1 Back/Next



**Figure 3-32: Next 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 objects that can be returned to appears.

### 3.2.2 New bookmark



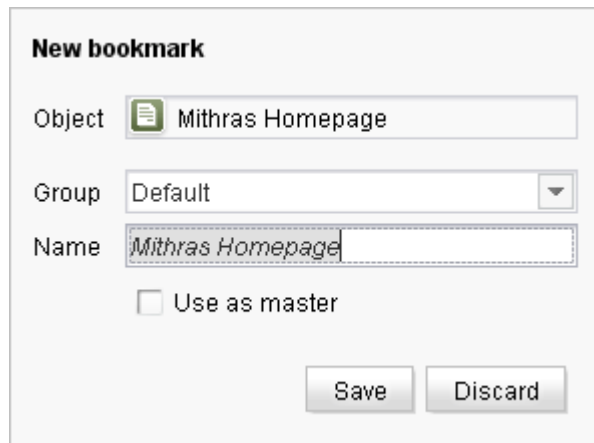
**Figure 3-33: Save and open bookmark**

With this function it is possible to save frequently used objects and to open them using the selection box next to this symbol. If this symbol is not visible, it can be shown again by widening the left-hand column (tree structure).



Bookmarks always relate to a specific project and are valid for only one specific user; several users of a project must each specify their "own" bookmarks.

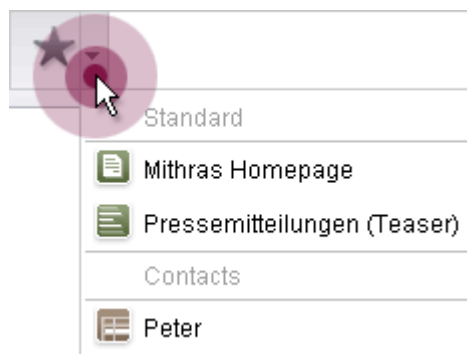
To create a bookmark, the required object is marked in the tree structure. Click the bookmark symbol to open the following window:



**Figure 3-34: New bookmark**

You can find more detailed information about creating bookmarks in Chapter 3.3.3.1 starting on page 101.

Use the arrow next to the bookmark icon in the tool bar to open a selection box where bookmarks and master copies are listed in groups.



**Figure 3-35: Open bookmark**

If a folder has been bookmarked, only this folder will be shown, but no subordinate folders or objects. Clicking the bookmark will display the object in question in the workspace.



### 3.2.3 Home



**Figure 3-36: Home**

Using this button the user can jump to his personalized project homepage from each position of the actual project. It provides a personalized overview about the bookmarks, open tasks and possible actions with icon and in the color of the respective store. For further information see section 3.6 page 124.

### 3.2.4 Refresh



**Figure 3-37: Refresh**

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 subtree 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.

### 3.2.5 Editing mode on/off



**Figure 3-38: Block editing of an object**

To make changes to an object, it is first necessary to switch on Editing 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, Editing 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** or context menu entry "Extras – Cancel editing" to quit Editing mode without adopting any of the changes made.

Depending on the settings in the server and project configuration, the user is prompted to make a comment on exiting Editing mode (see *FirstSpirit Manual for Administrators*).

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

The following options are possible:

**Forced comment:** The comment line can only be closed by clicking OK after the editor has entered a text. In other words, Editing mode only ends if a comment is entered:

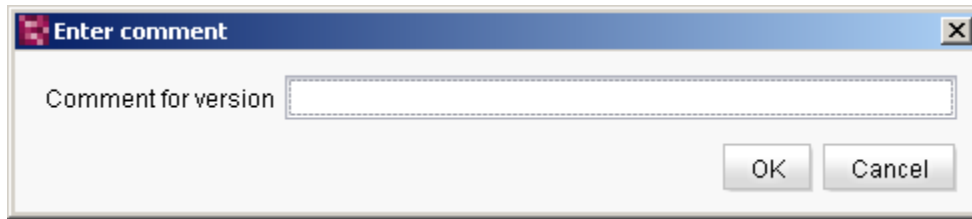


Figure 3-39: Forced version comment





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

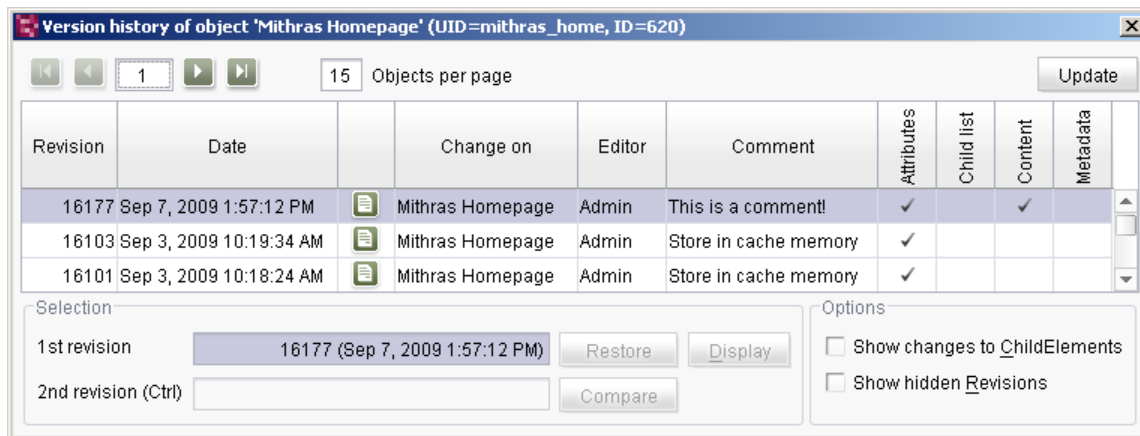


**Figure 3-40: Optional version comment**

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

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

The comments entered are also copied to the object's version history.



**Figure 3-41: Change comment in the version history**

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

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

### 3.2.6 Save



**Figure 3-42: Save**

This function is used to save changes to the object currently being edited. If the integrated preview is used, it is refreshed after each save so that the entries can be checked directly on the



website.

It is important to note that this is a kind of "temporary storage". An object is normally saved automatically as soon as Editing 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! In addition, it is possible to restore older statuses that were temporarily stored in this way.

### 3.2.7 Preview



**Figure 3-43: Preview**

If integrated preview for content is not enabled (see section 3.1.5.5, page 59), you can use this function to check how the finished website will look. Clicking this button links the content of the page to the layout and it generates a test version of the website. The page is then displayed in the preferred browser.

When you view a preview via the relevant context menus (e.g. in the Page Store, section 4.2.3, page 135) you can freely select from all languages and presentation channels assigned to the project.

In the Media Store, calling the preview function results in

- a preview of the image or the file in a new tab on the right-hand side of the window, if the integrated preview for media is used.
- a preview of the image or the file in an appropriate display program, if the integrated preview for media is not used.

Right-clicking on the preview icon opens the preview in an external web browser if integrated preview is enabled. Likewise, right-click to display the external preview of an image from the Media Store.



### 3.2.8 New



**Figure 3-44: Create new object**

This icon can be used to create new objects. The options available here always depend on the store or level you are currently in. If, for example, you are in the Site Store on a menu level, you can create a new page reference with this icon. This function is also accessible via the respective context menus and is described in greater detail in the relevant places.

### 3.2.9 Delete



**Figure 3-45: Delete**

Click the icon (or press the "Del" key) to delete the currently selected object or the currently selected subtree in the FirstSpirit JavaClient (see sections 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 (see section 3.2.9.7, page 87). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

Click the arrow next to the delete icon to display a list of all objects deleted during the current session. The deleted object or deleted subtree can be reinstated by selecting an object (see section 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 subtrees (see section 3.2.9.2 and 3.2.9.5).

### 3.2.9.1 Delete objects

To delete an individual objects, for example a section, highlight it 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:** Click this button to cancel the deletion. The selected element is retained in the project and the dialog is closed.

**OK:** Click this button to confirm deletion of the selected object. The object is removed from the project. The user can restore the deleted object during the current session by using the **Restore** function (see section 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:

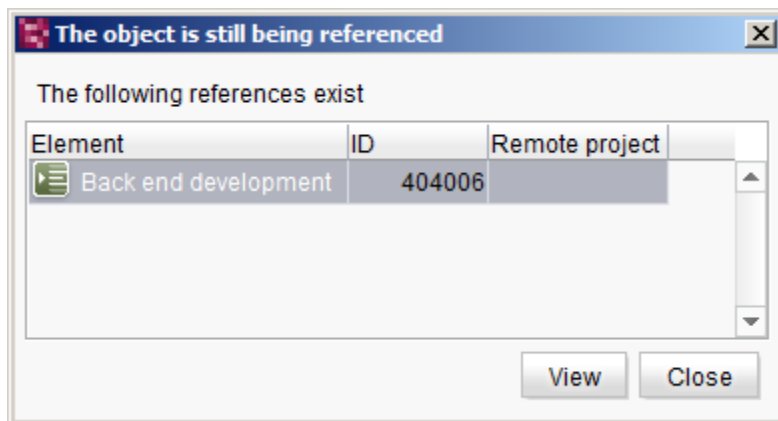


Figure 3-46: Display references for an object

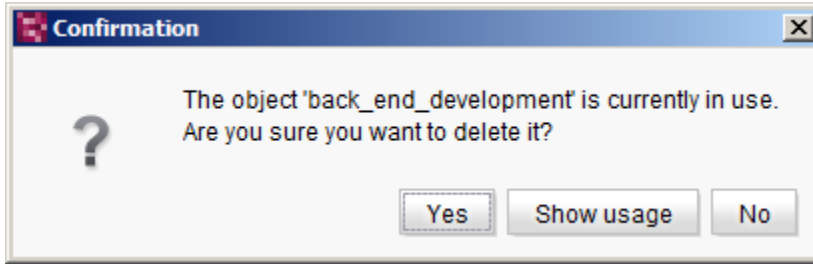
**View:** Deletion of objects still in use can result in invalid references within the project. Therefore, before an object can be deleted all its uses must be deleted. Within the table the user can select a use and click the **View** button to directly switch to the selected use and delete it if necessary.



**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 3-47: Delete object despite uses**

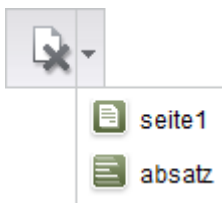
**Yes:** Click this button to remove the object from the project, although it is still being used within the project.

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

**No:** Click this button to cancel the deletion. The object is retained in the project and the dialog is closed.

### 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 3-48: 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 section 4.2.9 page 136).

### 3.2.9.4 Delete subtrees

Apart from simple deletion of objects, FirstSpirit also supports the deletion of subtrees. To delete a subtree 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:

**OK:** Click this button to confirm deletion of the selected subtree. All elements of the subtree 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 section 3.2.9.6).

**Cancel:** Click this button to cancel the deletion. The selected subtree remains in the project and the dialog is closed.

If elements selected for deletion within the subtree are still being used in the project they cannot be deleted (see section 3.2.9.1). The following dialog shows successful and unsuccessful delete actions on the server:

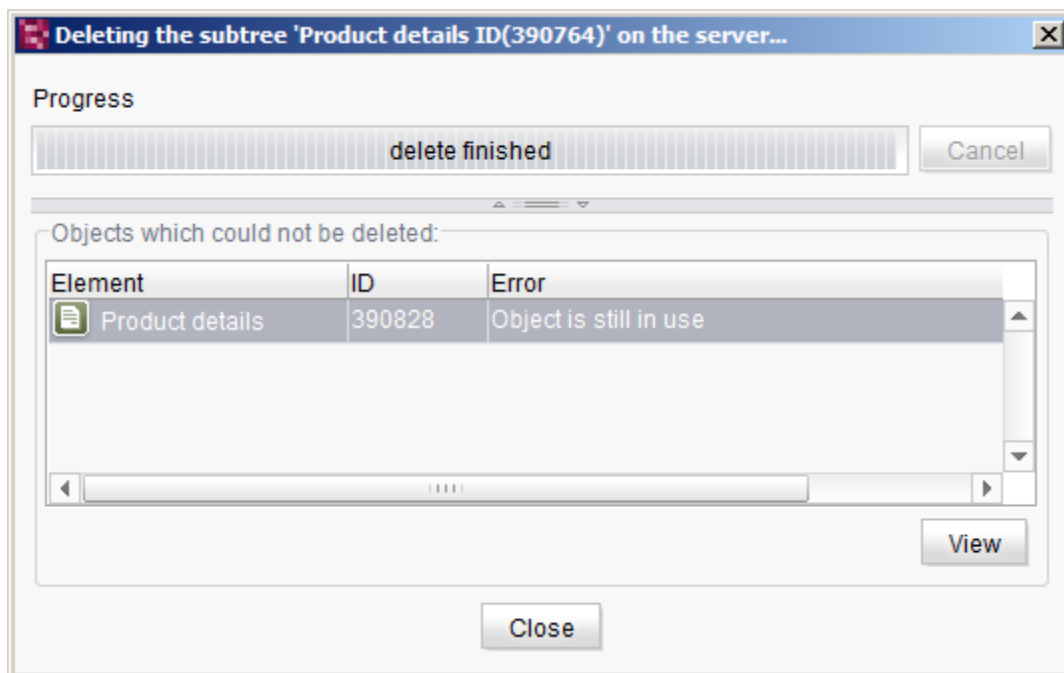


Figure 3-49: Delete a subtree



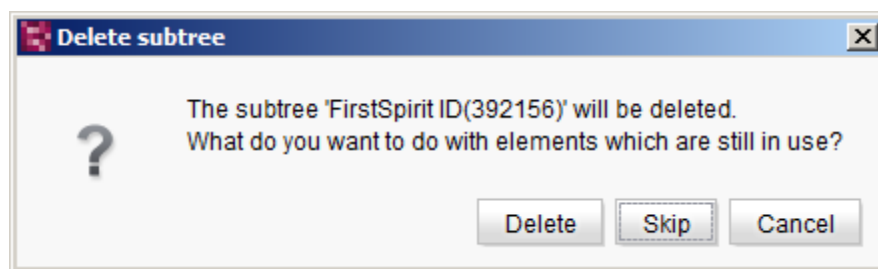
If a subtree is deleted in which at least one object is still in Editing mode, only the part of the content not being edited is deleted and the edited objects and the folder itself are kept.

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

Click the "View" button to display the selected element in the tree structure.

### 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 3-50: Delete a subtree (for administrators only)**

**Delete:** Click this button to remove all objects of the subtree from the project with a click, irrespective of whether they are still being used within the project or not. This can cause invalid references.

**Skip:** Click this button to delete in the same way as deleting subtrees without administration permissions. Only the elements of the sub-tree no longer being used in the project are deleted.

**Cancel:** Click this button to cancel the deletion. The selected subtree remains in the project and the dialog is closed.

### 3.2.9.6 Restore subtrees during a session

You can restore deleted objects in a subtree at any time within the individual stores via 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 section 4.2.9 page 136).



### 3.2.9.7 Connecting a workflow to the delete function

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 by 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 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 section 12.2 page 391).

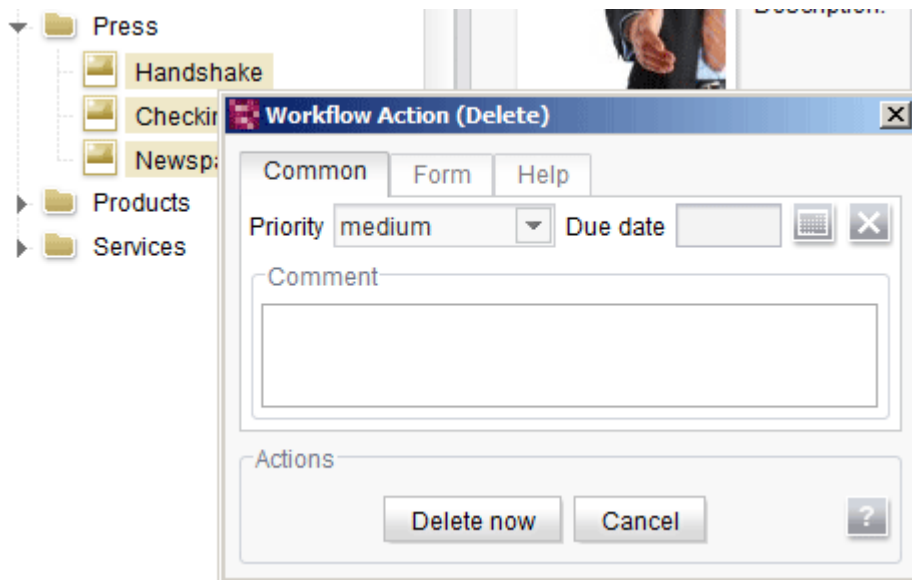


Figure 3-51: Deleting several objects by means of a workflow

For further information about the deletion workflow, see *FirstSpirit Manual for Developers (Basics)*.





### 3.2.10 TabView



**Figure 3-52: TabView**

This function can be used to display an overview of the workspaces in the AppCenter area. (For details of the extended workspace view, see section 3.5.3 page 123.)

### 3.2.11 Show task list



**Figure 3-53: Show task list**

This function can be used to show your own task list. (The task list is described in detail in section 11.1, page 305.)

### 3.2.12 Online Help







**Figure 3-54: Opening Online Help**

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



### 3.3 The vertical tool bar

On the left-hand edge of the screen, there is a vertical tool bar that can be used to control which content is displayed in the left-hand column. The functions in the left-hand area can be grouped into four different groups:

-  Search (see section 3.3.1, page 88)
-  Explore (see section 3.3.2, page 99)
-  Organize (see section 3.3.3, page 101)
-  Multisite Management (see section 3.3.4, page 112)



### 3.3.1 The global search ("Search")

You can use the form field in the top left-hand part of the JavaClient to start a global search. The search results are displayed directly in the left-hand area of the JavaClient where the project's tree structure is normally displayed:

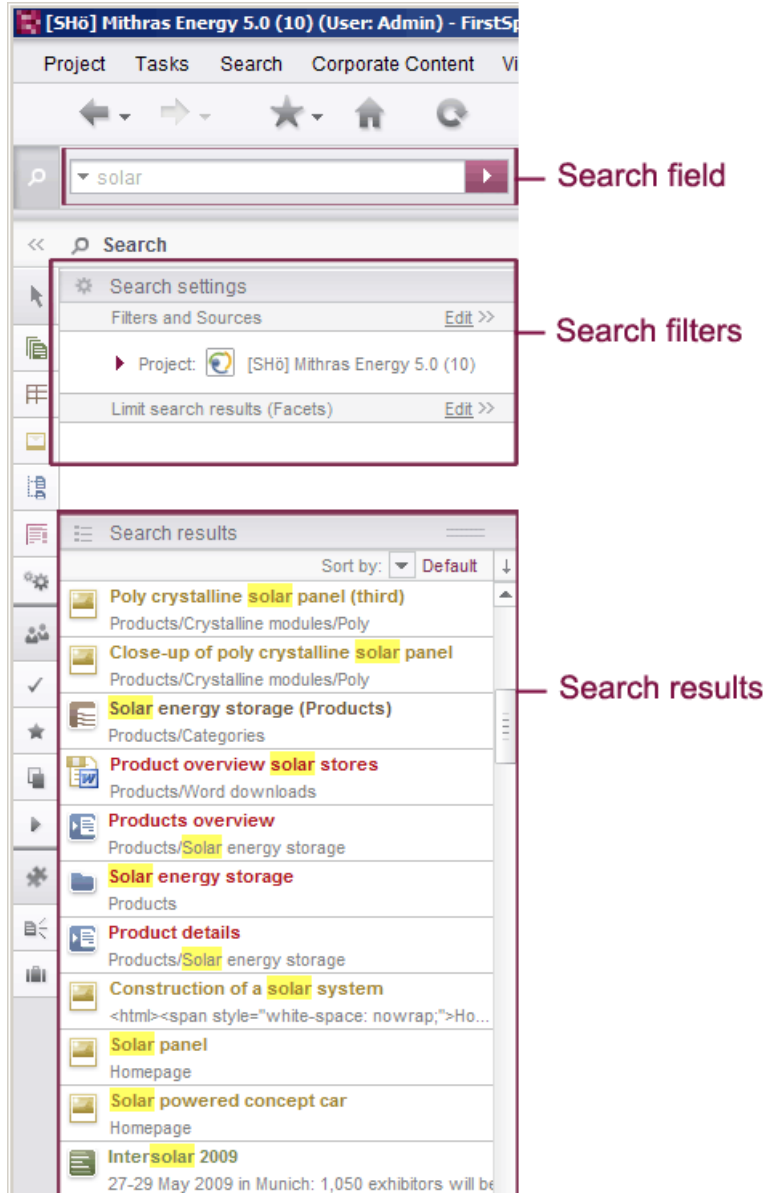


Figure 3-55: Global search



### 3.3.1.1 Search field

▼ With this icon in the search field, search terms for which searches have recently been performed (including those before the current session), can be displayed in a list. Select a search term from the list to perform a search for this term. Use **Delete last searches** to delete the list.

The search begins while the first characters are being entered in the input field and shows terms, which begin with the characters entered and are available in the project (auto completion). The number of hits for the suggested term is displayed in brackets. Click a suggestion to copy the term into the search and the corresponding hits are displayed in the results list (see Chapter 3.3.1.3 page 92).



If you want to search for the term, as it was entered, without using a suggestion, you can start the search with ENTER or by using this icon.

✕ Use this icon to delete the search term from the input field.

A full text search is performed, which is not case sensitive. Search terms can therefore be entered in upper and lower case letters. If several search terms are entered, the search for these terms is based on an AND operation; only objects that contain all the search terms entered are displayed in the results list. Here it must be noted that, because of the search technology used, search terms with hyphens (-) cannot be used. Instead, the words linked by hyphens should be entered separately, e.g. know how or knowhow instead of know-how.

Not only object names are searched through, but also content (e.g. of pages, datasets or media); in the case of media, text in the "Description" field is also searched through.

If more than 500 hits are found for a search, a corresponding message appears: "The search returns at least 500 hits. Do you still wish to continue?" If **Yes** is selected, the search is continued. If **No** is selected the search is cancelled; the first 500 hits are then available in the results list.



If the search has been started, it can be stopped by clicking this icon.

If the search is finished, this is indicated in the status bar.



### 3.3.1.2 Non-text search

Alternatively, FirstSpirit objects can be dragged into the search field with drag-and-drop (for details of drag-and-drop functionalities, see also Chapter 11.4 page 310 ff.), for example

- nodes from the tree structure (pages, sections, media, templates, etc.)
- workspace tabs
- individual input components with content or only their content from an open workspace
- elements from the integrated preview

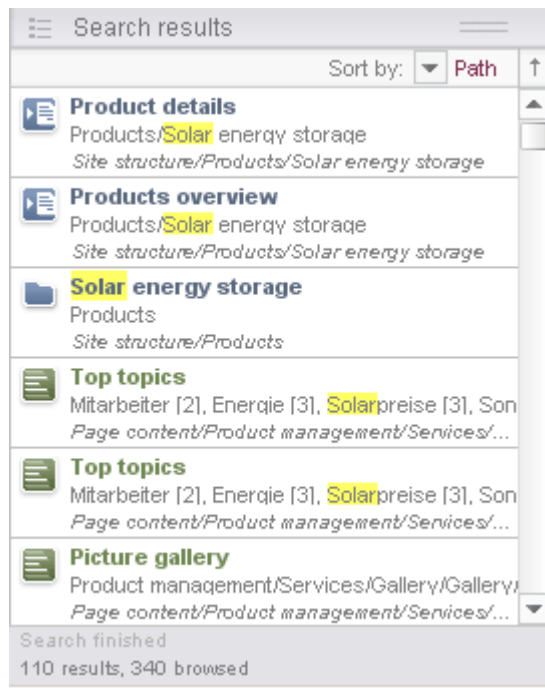
Depending on the object, the search then returns, for example, the following:

- Uses of the dropped **object** in the project, e.g.:
  - Where in the project a medium from the integrated preview is used or maintained?
  - Where in the project is a medium from the tree structure, a workspace or the integrated preview being used?
- Uses of values saved in **input components**, e.g.:
  - Where are right-aligned pictures used?
  - Is a heading or another text being used anywhere else?
  - Which other datasets belong to a category?

### 3.3.1.3 Search results

The search hits are displayed beneath the input field (see Chapter 3.3.1.1 page 91) and the Search Settings (see Chapter 3.3.1.4 page 94 ff.).






**Figure 3-56: Search results**

They are listed in the left-hand column with the object icon, title and a second text row. By default, the title is the name of the object, the second row is the object's path; however, it can also be text parts of the respective object. Further information about the object can be displayed in a third text row, if a sorting criterium is selected via **Sort by** (see below). If a search hit is a picture from the Media Store, it is displayed on the right as a preview image (thumbnail). If the mouse cursor is held over the search result the thumbnail is displayed enlarged. Depending on the project configuration, pictures can also be displayed for other object types (pages, sections and datasets).

The search term is highlighted in color.

By default, the results are displayed according to their relevance. They can be sorted as follows:

 **Sort by:** Use this icon to sort the search result by the following criteria:


- editor (alphabetically)
- location (alphabetical, incl. Store name)
- change date
- criteria selected under "Limit search results" (see Chapter 3.3.1.5 page 96)

The original sorting can be restored by selecting the option "Default".



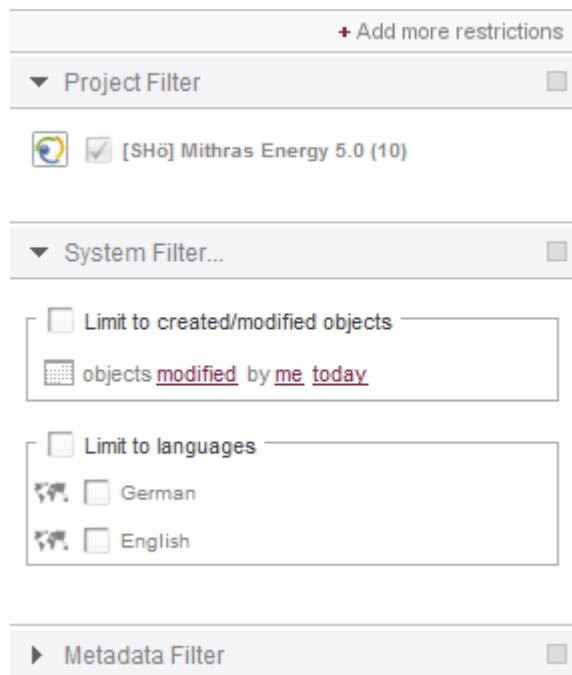
↓ **Descending / Ascending:** Use this icon to display the search results in reverse order.

The search results can be limited to certain criteria in the "Search Settings" area (see Chapter 3.3.1.4 page 94 and Chapter 3.3.1.5 page 96).

The search dialog can be left open, without restricting further work in the JavaClient. In this way, for example, it is possible to drag-and-drop the search results into the JavaClient, e.g. the editor can drag a medium directly from the search results into the picture input component of a workspace (see Chapter 11.4 page 310). The search results are displayed until a new search is performed. After using the icons to change to a Store in the left-hand area or another area, click the Search icon  to switch back to the search results.

### 3.3.1.4 Search settings (filters and sources)

The **Filters and Sources** entry lets the user filter search results. This is done in the Edit window that opens as soon as the user clicks on Edit in the Filters and Sources title row.



**Figure 3-57: Search settings – "Filters and Sources"**

- **Project Filter**  
Here the user can specify whether to search in the current project or in a remote project.
- **System Filter**  
Here the user can specify system element search limitations:



Limit to created/modified objects: here the user can specify whether the search results are filtered by objects that

- were created or modified ("created" or "modified")
- within a certain period of time ("today", "this week", "last week", etc.)
- by a particular user of the project (e.g. "Admin").

Limit to languages: here the user can specify whether to filter the search results so that only the results for a particular project language are output.

References / Elements: here the user can specify whether the search results are filtered by objects that have

- invalid references (e.g. to a project that has since been deleted),
- external references,
- not released references (e.g. pages in the Site Store that do not reference released pages from the Page Store), or
- not released elements.

Show all tasks: here the user can specify whether to filter the search results by objects for which a workflow is active and the user can select which user is responsible for this task ("me", "user name", "group name").

Translated / not translated elements: here the user can specify whether the search results are filtered by objects for which a translation

- is available ("translated", "not translated") for particular project languages
- ("DE", "EN", "all languages", "any language").

▪ **Metadata Filter**

Here the user can search the project metadata for particular text: The search procedure is similar to the "Editor search" of the "Metadata search" in the "Search" menu (see section 3.1.3.5, page 47).

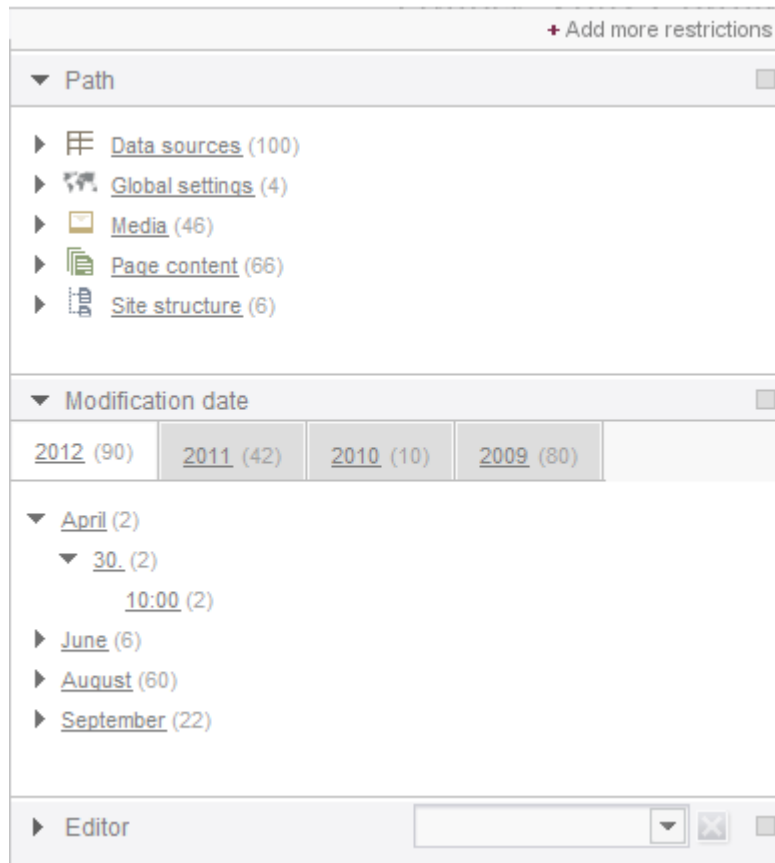
■ Active filters are displayed by a blackberry-colored box next to the filter on the right. The selected filters are also visualized in the "Search Settings" area. Each filter can be disabled by the X next to the filter.





### 3.3.1.5 Search settings (limiting the number of hits ("Facets"))

With the **Limit search results (Facets) / Edit >>**



**Figure 3-58: Search settings – "Facets"**

entry, the search results can be filtered, namely by

- **Path**  
In this area the search results can be limited to a Store area and folders that exist within the Store areas.
- **Modification date**  
In this area the search result can be filtered by change date, i.e. by the date on which the object was last edited. The year is selected first, then the month, day and time. The time can be limited to the nearest hour.
- **Editor**  
The required user can be selected from this dropdown menu. Only search hits edited by the selected user are then considered.



In all displays of search results, the number of search hits is displayed in brackets.

■ Active filters are displayed by a blackberry-colored box next to the filter on the right. The selected filters are also visualized in the "Search Settings" area. Each filter can be disabled by the X next to the filter.

In addition, **Add more restrictions** can be used to enable the following filters:

- **Created by**

The required user can be selected from this dropdown menu. Only objects created by the selected user are then considered. (It is possible that this information may not be completely available for projects originating from older versions of FirstSpirit.)

- **Type**

The search result can be filtered by FirstSpirit object type in this area, e.g. by

- Content sources
- Datasets
- Files
- Media
- Pages
- Page references
- Menu levels ("Site Store folders")
- Images
- Sections
- Section references
- Page templates
- Table templates
- Project settings

- **Released by**

The required user can be selected from this dropdown menu. Only objects released by the selected user are then considered.

- **File type**

If a search hit is a medium, the MIME type (*Multipurpose Internet Mail Extensions*) can be selected here (e.g. "text/plain", "image/jpeg", "application/msword", etc.).

- **Schema**

If a search hit is a hit from the Content Store, the underlying Database Schema can be selected in this area.

- **Creation date**

In this area the search results can be filtered by creation date, i.e. by the date on which the object was created. The year is selected first, then the month, day and time.



- **Release date**

In this area the search result can be filtered by release date, i.e. by the date on which the object was last released. The year is selected first, then the month, day and time.

- **File size**

If a search hit is a medium, the file size can be selected here. The following values are available to choose from

- EMPTY (no data stored)
- KB100 (files up to 100 KB in size)
- MB (files up to 1 MB in size)
- MB10 (files up to 10 MB in size)
- MB100 (Files up to 100 MB in size)

- **Release status**

In this area the search result can be filtered by release status. The following values are available to choose from

- released
- not released
- in workflow

- **Workflow**

Use this option to filter search hits by workflow.

- **Table**

If a search hit is a hit from the Content Store, the underlying table can be selected in this area.

- **File name extension**

If a search hit is a medium, the file name extension can be selected here (e.g. png, jpg, pdf, doc, swf, etc.). The file name extension does not necessarily reflect the file type. Therefore, the "File type" filter should be selected for filtering by file type (see above).

- **Metadata**

In this area you can select whether only search hits with ("1") or without ("0") metadata are to be displayed.

- **Release lock**

In this area you can select whether only search hits, which are located in a workflow and locked ("0") or not ("-") are to be displayed.

All criteria can also be combined with each other. With each criterion, the results set is further limited. After the required criteria have been selected, the search hits that simultaneously fulfill all criteria are displayed (intersection).







The settings can be collapsed or expanded using the icons.



If the filter settings have been changed, which affect an already displayed search result, the filters can be applied by using the "Repeat search" link.



### 3.3.2 The tree structure ("Explore")

This group includes the six different stores in FirstSpirit:

-  Page Store: see section 4, page 122
-  Content Store: see section 5, page 150
-  Media Store: see section 6, page 177
-  Site Store: see section 7, page 227
-  Template Store: see section 8, page 257
-  Global content area: see section 9, page 259

#### 3.3.2.1 Navigation using the keyboard

You can navigate through the tree structure using the cursor keys. The "up" and "down" cursor keys move you to the next node displayed in the tree structure above or below the current node. The "right" and "left" cursor keys expand or contract nodes beneath the current node (e.g. sections beneath a page in the Page Store).

A colored background (e.g.  **Crystalline modules**) indicates that the content for this node is currently displayed in the workspace. If the user navigates using the keyboard, the current selection is initially retained; only the focus within the tree view (visualized by a colored frame, e.g.  **Crystalline modules**) 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 focused 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.



### 3.3.2.2 Multiple selection

The following applies to multiple selection within the tree: All selected elements are assigned a background color, which depends on the respective Store. The last selected element is also focused and therefore has a black frame. If the focused element lies outside the multiple selection (e.g. in the case of navigation using the keyboard) it is marked by a colored frame (see Figure 3-59).

Multiple selection using the keyboard is also possible (Shift + ↑ or Shift + ↓). Here, as with navigation using the keyboard (see section 3.3.2.1, page 99), the focus is changed first (colored frame). The elements of the multiple selection are not selected (and highlighted with a colored 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.

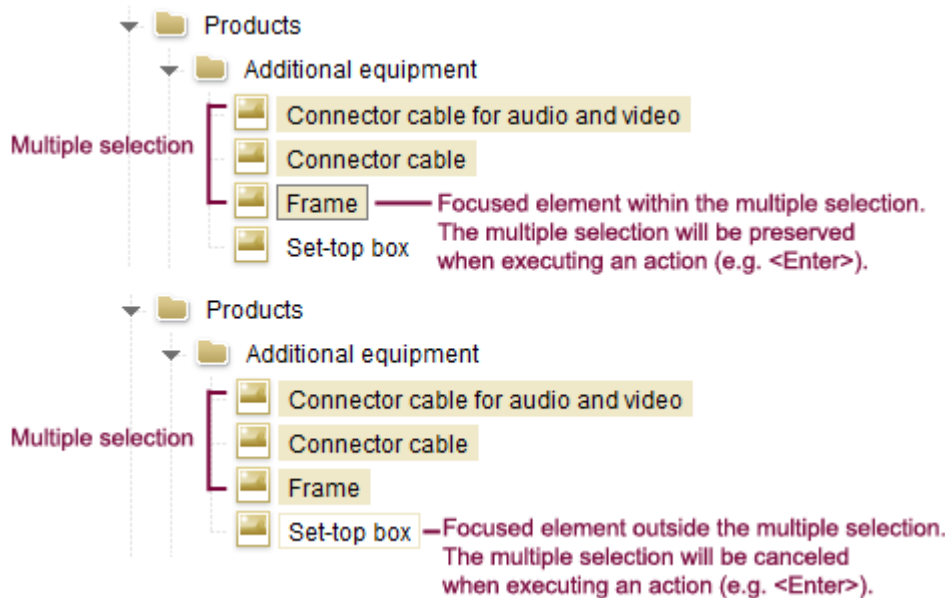


Figure 3-59: Multiple selection in the tree

On switching the workspace (with enabled tree synchronization), 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.




Multiple selection can be used in the **Content Store** too. Here, selection is not made in the tree




structure but on dataset level. More than one dataset can be selected by pressing the <Ctrl> and <Shift> key simultaneously. When **deleting** more than one simultaneously selected datasets only one confirmation prompt will be shown for all datasets. If you confirm it all selected datasets will be deleted.

### 3.3.3 The user-defined view ("Organize")

There are four views within the "Organize" area which can be displayed for each user using the related icons:

-  Bookmarks (see Chapter 3.3.3.1 page 101)
-  Actions (see Chapter 3.3.3.2 page 108)
-  Clipboard (see Chapter 3.3.3.3 page 109)
-  Task list (see Chapter 3.3.3.4 page 111)

The order of the Organize icons can be changed and they can be combined to a group.

-  Group with two views

If you would like to change the order please click the icon and drag it to the desired position within the Organize area. You can also drag an icon onto another by simultaneously pressing the STRG key. The both icons will then be displayed as a group in one field. Maximally four icons can be combined within one group.

#### 3.3.3.1 Bookmarks

Most people are familiar with bookmarks in Internet browsers. They are a tried-and-tested method for users to quickly access important and frequently-used websites.

Bookmarks enable editors to create their own working environment in the FirstSpirit JavaClient. This can prevent the need to switch back and forth between the stores. Bookmarks can be assigned to individual nodes in the tree structure, to datasets and also to search queries.

Most bookmarks can also be tagged as so-called **master copies**. Master copies can be used to quickly create copies of frequently used FirstSpirit objects, including all their subordinate



elements and entered contents.

Bookmarks are created and edited using the horizontal FirstSpirit tool bar (see section 3.2.2, page 76).

### 3.3.3.1.1 Creating bookmarks and master copies

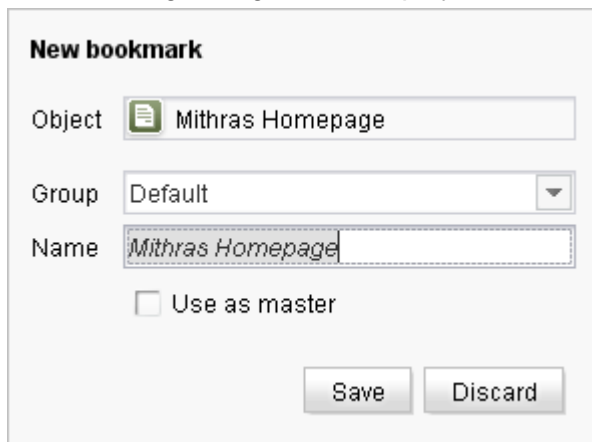


Bookmarks are created as before by using this icon in the JavaClient tool bar. Master copies are also created using this icon. If the icon is not displayed it can be shown by widening the left column (tree structure).

All object types can be bookmarked on principle. In the case of folders, subordinate objects can subsequently also be invoked in the left-hand column of the JavaClient, but this only applies to the objects located one level below the folder concerned (see Chapter 3.3.3.1.3 page 104, section on the vertical tool bar). Master copies can only be created for specific FirstSpirit object types.

Bookmarks and/or master copies can be created for data records by way of the context menu option "New bookmark".

The following dialog will be displayed:



**Figure 3-60: Creating a new bookmark**

**Object:** This field displays the selected object the bookmark and/or master copy is to be created for, including the name and object icon.

**Group:** Bookmarks and master copies can be filed in user-defined groups which need to be created beforehand (see Chapter 3.3.3.1.2 page 103). The desired group can be selected from this drop-down list. The group "Default" is provided by default



and cannot be deleted. A bookmark or master copy can only be allocated to one group. The allocation to a group can be changed later on as required (see Chapter 3.3.3.1.4 page 106).

**Name:** This field can be used to assign a name to the bookmark or master copy by which it can be located later. It contains the name of the object by default, but this can be changed in this dialog or also later on (see Chapter 3.3.3.1.3 page 104, "vertical tool bar" and Chapter 3.3.3.1.4 page 106). Names can be used several times over.

**Use as master copy:** If this checkbox is activated, new objects can be created on the basis of the current object ("Master copy"). Master copies created in this manner can subsequently also be accessed in the JavaClient using the context menu "New" (see also Chapter 3.3.3.1.5 page 107). Master copies can also be used in the WebClient (see also the *WebClient Manual for Editors*). Not all object types can be used as master copies and the checkbox may be disabled for this reason.


**Save** serves to save the bookmark for the selected object along with its name and group allocation. It is then available at the arrow next to the bookmark icon in the tool bar and the corresponding icon in the vertical tool bar.

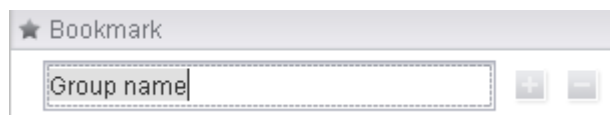
If the **Discard** button is clicked, no bookmark will be created for the selected object and the dialog will be closed again.

Several bookmarks can be created for one object.

### 3.3.3.1.2 Creating and deleting groups

Groups help to easily structure bookmarks in FirstSpirit clients.

 Create bookmark group: this icon is used to create new groups. The combobox will turn into a text field:




**Figure 3-61: Creating a bookmark group**

This field can be used to enter a suitable name for the group of bookmarks. The group can then be included in the list of groups by pressing ENTER.



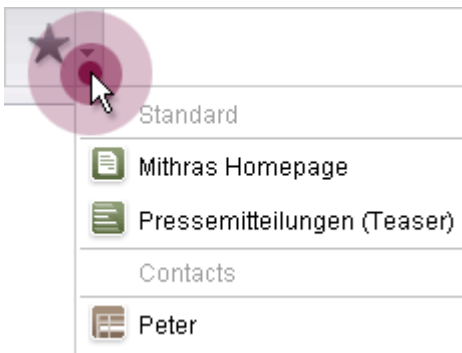


 Delete bookmark group: this icon can be used to delete the group currently shown in the drop-down list. All the bookmarks included in the group will also be deleted at the same time. The "Default" group cannot be deleted.

### 3.3.3.1.3 Displaying and opening bookmarks and master copies

Bookmarks, master copies and the corresponding objects can be called up and opened in the workspace at the following locations in the JavaClient:

- The arrow next to the bookmark icon in the tool bar:



**Figure 3-62: Bookmarks / Tool bar**

This is where the bookmarks and master copies are listed in groups. If a folder has been bookmarked, only this folder will be shown, but no subordinate folders or objects. Clicking the bookmark will display the object in question in the workspace.

- "My bookmarks" in the project entry page (icon ):

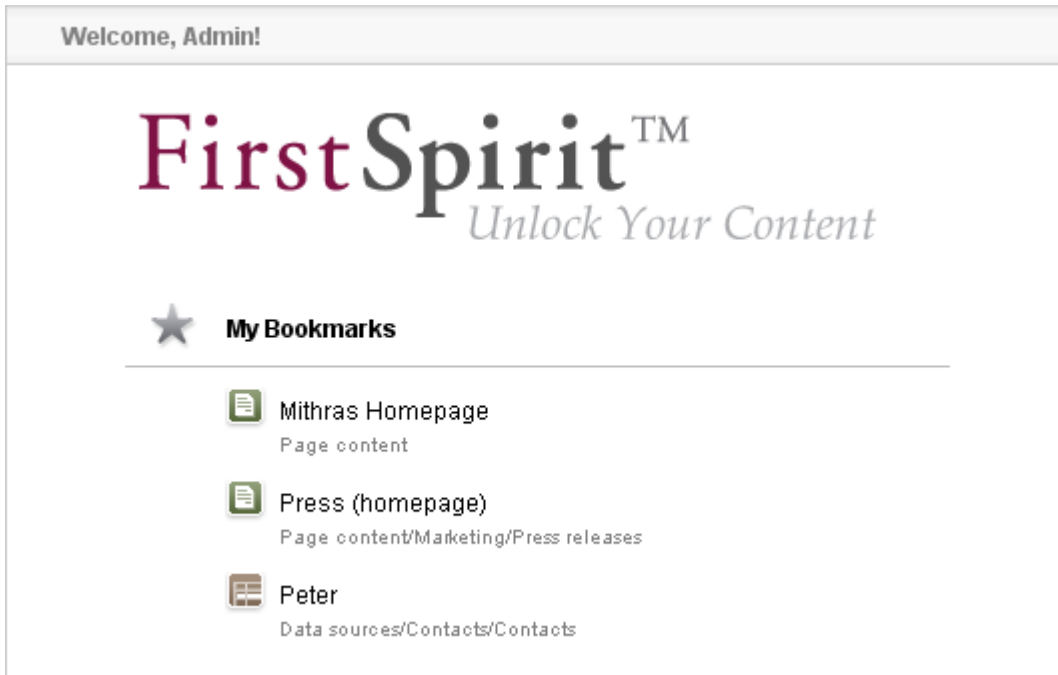


Figure 3-63; Bookmarks / Project entry page

If a folder has been bookmarked, only this folder will be shown, but no subordinate folders or objects. Clicking the bookmark will display the object in question in the workspace.

- The corresponding icon in the vertical tool bar (icon ):



Figure 3-64: : Bookmarks / Vertical tool bar

This is where the bookmarks are displayed in groups along with their name, path and/or supplementary text and a preview image. Master copies (see Chapter 3.3.3.1.1 page 102, option "Use as master copy") are identified by an asterisk at the object icon. The respective object can be opened in the workspace with one click on an entry.

To view bookmarks or master copies from other groups the desired group needs to be selected from the drop-down list.



If a folder has been bookmarked, subordinate objects will also be shown here, but only the objects **one** level below the folder concerned. Subordinated elements can also be displayed in the workspace by clicking them.

The name of the bookmarks or master copies can be changed in this view by double-clicking it.

Bookmarks and master copies are listed in their order of creation by default, but this order can be changed by means of drag and drop.

If the mouse cursor moves across a bookmark or master copy, the icons described in Chapter 3.3.3.1.4 on page 106 will be shown.

#### 3.3.3.1.4 Editing and deleting bookmarks and master copies

The bookmarks and master copies opened by way of the vertical tool bar can be edited using the following icons:

✕ Delete: use this icon to delete the bookmark or master copy. This function is only available for objects that are directly tagged as bookmarks or master copies, and hence not for subordinate objects. If a folder has been tagged as a bookmark or master copy, for example, the icon will only remove this tag from the folder itself, but not from its subordinate folders or objects.

» Edit: this icon will open a dialog like the one in Figure 3-60, where the name, group allocation, and the "**Use as master copy**" option can be changed. This function is only available for objects that are directly tagged as bookmarks or master copies, and hence not for subordinate objects. If a folder has been tagged as a bookmark or master copy, for example, the icon will only edit this tag from the folder itself, but not from its subordinate folders or objects. The dialog can be closed without making any changes by clicking "Discard".



### 3.3.3.1.5 Using master copies

Bookmarks where the option "Use as master copy" is activated (see Chapter 3.3.3.1.1 page 102) can be accessed

- via the "New" option in the context menu of some nodes in the tree structure



Figure 3-65: : Function "New" in tree node context menu

- in the dialog "New" / "Insert section" or "New" / "Insert new page" (see Chapter 4.1.1 page 129)
- at the icon in the horizontal tool bar

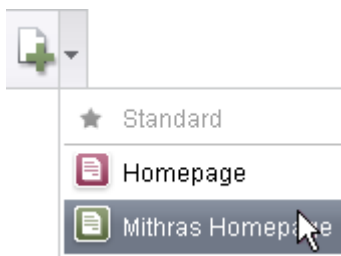


Figure 3-66: Funktion "Neu" in der horizontalen Symbolleiste

genutzt werden

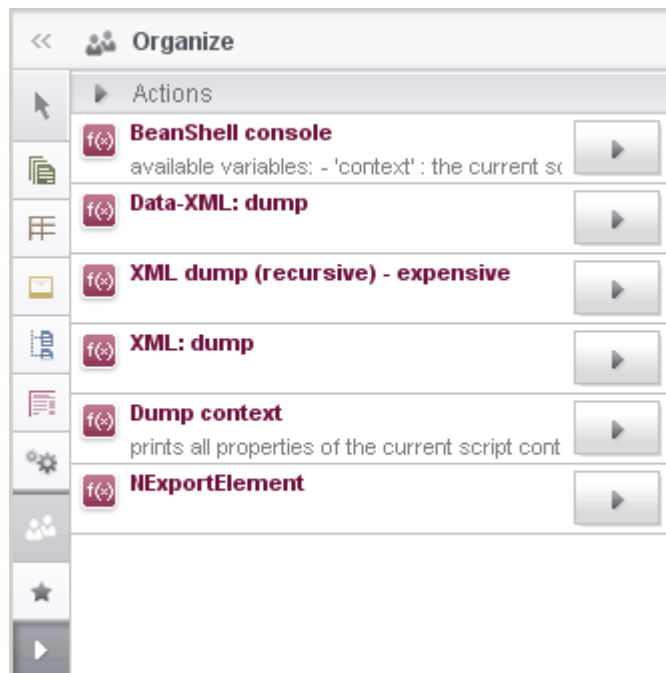
Which master copy is being displayed always depends on the store and/or node one is currently located in. If one is located in a page in the **Page Store**, for example, all folders and pages in the Page Store and page templates and pages of the "Global Content Area" that are tagged as master copies will be displayed. In the content areas of pages, only those sections that are allowed for the respective page will be shown as master copies. In a menu level in the **Site Store**, the bookmarked menu levels and page references will be shown.

If a master copy is selected, an exact copy of the object tagged as a master copy will be newly created at the selected position in the tree structure, including all subordinate elements and entered contents. If a Page Store folder has been tagged as a master copy, the "New" function, for example, will create a copy of the folder including all subfolders and subordinate objects. When selecting page templates (from the Template Store), the "New" function will create a new page on the page template which is tagged as master copy.




The name assigned to new objects which are based on master copies is usually the **display name** of the object serving as a master copy. For pages and sections that are based on master copies from the Template Store, however, the **reference name** of the template is used.

### 3.3.3.2 Actions



**Figure 3-67: Actions**

In this area the available menu and context menu scripts are listed which are available for the object which is activated in the editing area. Scripts enable pre-programmed actions or calculations to be executed. For each shown script you can start the respective action by clicking on the icon  behind the script. All actions which are started from this position will be carried out context-related.

Depending on the configuration by the project developer, actions also can be started by means of

- the menu "Extras" / "Execute script" (Chapter 3.1.6.11 page 72),
- the project entry page (Chapter 3.6 page 124) or
- context menus (e.g. Chapter 4.1.12 page 134)



### 3.3.3.3 Clipboard

The clipboard can be used to hold objects, so that they can be used elsewhere later; not only FirstSpirit objects, e.g. pages, page references, image, but also datasets, sections, individual input components or texts, as well as files from the local desktop computer, e.g. images and office files, can be temporarily saved on the clipboard. The editor can use the clipboard as a "collection tank", in which they can clearly and centrally collect materials and content needed for operations to be carried out later.

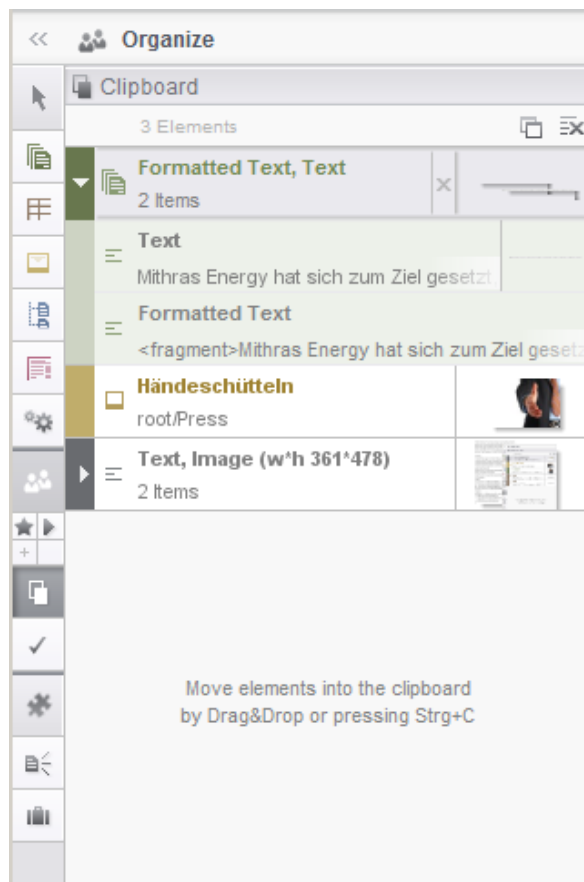





Figure 3-68: Clipboard







The clipboard remains open in the left-hand area until another area is selected using the vertical icon bar or the column is closed using the  icon or the separator bar. However, this does not delete the contents of the clipboard and they are still available after changing to another area; the contents remain available until the JavaClient is exited or the clipboard contents are deleted manually.

The objects are listed on the clipboard with a descriptive text, a thumbnail and if applicable an icon and the color of the respective Store. They are included in the list by <Ctrl> + C or the corresponding (context) menu function in FirstSpirit JavaClient or in third party software, or by drag-and-drop. For example, you can drag-and-drop nodes from the tree structure, input components and their content from an open workspace, elements from the integrated preview, etc. into the clipboard. In general, it is also possible to select several objects at once with the <Ctrl> or <Shift> depressed key and then drag them onto the clipboard. The paste icon  indicates that the object(s) can be dropped onto the clipboard. It is also possible to drop them onto the clipboard icon in the vertical icon bar. If the clipboard icon in the vertical icon bar is flashing, this indicates that an object has been copied onto the clipboard, even if the clipboard itself is not open.

Several objects can include several entries on the clipboard, e.g. copied text from an input component can be represented as text and as a picture. The entries can be expanded using the  icon on the left-hand side. How many objects are on the clipboard is displayed above the list (e.g. "2 elements").

If the mouse cursor is held over an entry, after a short time a drop-down box opens with an enlarged display of the object.

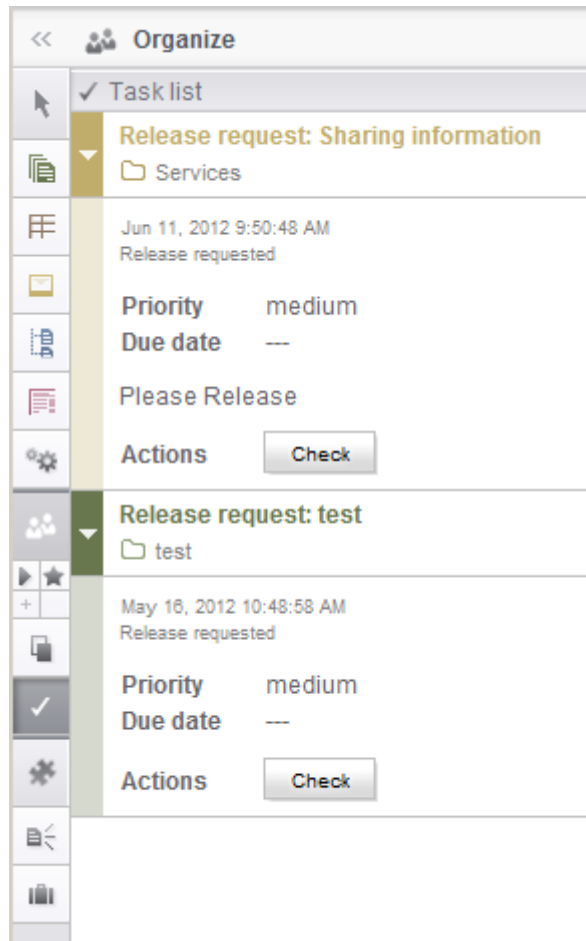
-  (<Ctrl> + <Shift> + V): Use this icon to open the clipboard in a separate window, which remains in the foreground.
-  (<Ctrl> + <Alt> + <Shift> + C): Use this icon to delete all the entries on the clipboard.
- : Use this icon to remove the respective entry from the clipboard.
- : If the entry is a FirstSpirit object, use this icon to switch directly to the object. It is opened in a tab in the workspace where it can be further edited.

The content of an entry can also be used elsewhere in the JavaClient by means of drag-and-drop.



3.3.3.4 Task list 

Upcoming tasks can be displayed in the Organize area:



**Figure 3-69: Task list**

Here, they are listed by date of start or transition with the following information:

- name of the workflow
- name of the element
- color of the object
- path

With double clicking an entry you can open the respective object in the workspace.





When you open the task you will get in addition the following information:

- start time
- status
- priority
- due date
- comment
- action button / next transition

For further information on the task list and on tasks / workflows please see Chapter 11.1 page 305 and Chapter 12 page 390.

### 3.3.4 The Multisite Management area

Various FirstSpirit modules such as CorporateContent or ContentTransport can be displayed in the Multisite Management area. These modules are license-dependent and are described in separate documentation.

## 3.4 FirstSpirit AppCenter/Integrated preview

The right-hand area of the screen contains the FirstSpirit AppCenter.

"FirstSpirit AppCenter" refers to the idea of seamlessly integrating third-party applications into the FirstSpirit editing environment. The FirstSpirit AppCenter makes a certain range within the editing system available within which independent applications that are not part of FirstSpirit can run (known as "AppCenter applications").

Examples of AppCenter applications include integrating Microsoft Office (see section 3.1.5.11, page 63 and section 6.7.1, page 213) or functions for integrated image processing (see section 6.6.3, page 201). The integrated web browsers Mozilla Firefox and Microsoft Internet Explorer are also AppCenter applications; these are called the **Integrated preview**. All these AppCenter applications were implemented by e-Spirit as product components. However, AppCenter applications can also be customer-specific. For more information about the FirstSpirit AppCenter, see *FirstSpirit Release Notes 4.2R4*, "FirstSpirit AppCenter" section.

The display on the right-hand part of the screen therefore depends partly on the project developer's specifications and partly on the project administrator's settings. In addition to this, the editor can also influence the functions in this area, via the menu items "Integrated preview," "Browser engine," "Office engine," and "Graphic engine" in the "View" menu of the FirstSpirit



menu bar (section 3.1.5, page 55).

### 3.4.1 Integrated preview for content

If the "Integrated preview – use for content" option is enabled in the "View" menu, the editor sees a direct WYSIWYG preview in the right-hand part of the screen for the content that the editor is editing in the FirstSpirit JavaClient. Apart from the form-based workspace in JavaClient, at the same time the content of the page being edited, the section being edited or the dataset is displayed in the integrated browser:

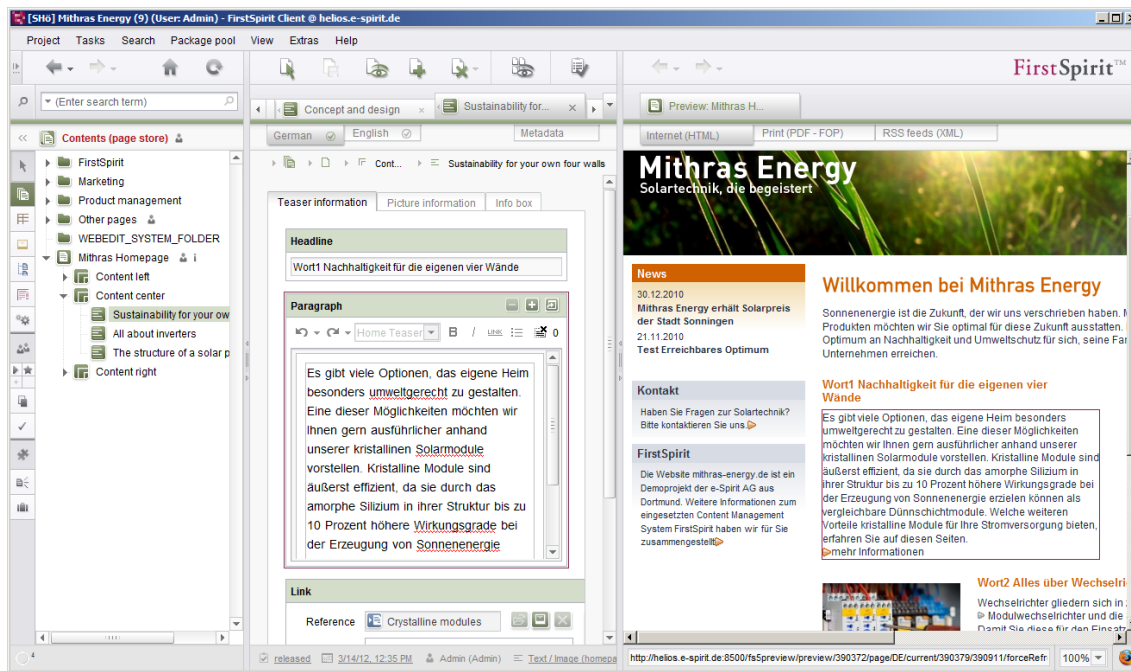
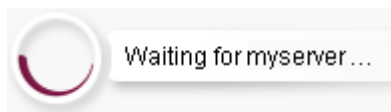


Figure 3-70: Display of the WYSIWYG preview

All editorial changes are automatically updated in the preview area (after saving).



This loading icon indicates recalculation; at this time, the preview is displayed toned down for the duration.

The integrated preview can be used in different stores: Within the **Content and Site Stores**, the editorial content is displayed in the respective output channel, for example as an HTML page or as a PDF.

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



Store is displayed in the integrated preview.

The **changeover between the various presentation channels** takes place simply via tabs within the integrated preview. This means it is no longer necessary to switch between JavaClient and web browser.



*The display of an output channel in the integrated preview may require 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.*

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)

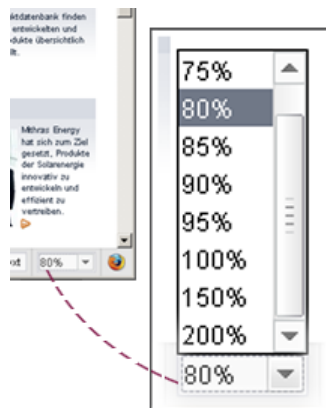
And, depending on the area that is clicked in the integrated preview:

- 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 dataset (click)
- Edit dataset (shift + click)

The integrated preview is configured using the "View" menu (see section 3.1.5.5, page 59).

In addition, a scaling factor of 50% to 200% can be defined in the bottom part of the preview (see Figure 3-71). 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 3-71: 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.*



### 3.4.2 Integrated preview for media

In addition to the content from the Page and Content Stores, the integrated preview can also be used to display and, in part, edit media – depending on the project administrator's specifications. To do this, the "Integrated preview"/"use for media" option must be enabled in the "View" menu.

Third-party applications integrated in the JavaClient enable editing of media, e.g.

Media can be displayed in the following way:

- **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 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 not a 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, **not** mixed mode.*



*If changes are made to documents, e.g. Microsoft Excel worksheets, which are not in FirstSpirit Editing mode, 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 color (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 an image or file in the FirstSpirit Media Store.*

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



### 3.4.3 FirstSpirit AppCenter

AppCenter applications are always displayed using an application appropriate to the format. Both the FirstSpirit Online Help and other file formats can be displayed in the FirstSpirit AppCenter, as well as functions for integrated image processing.

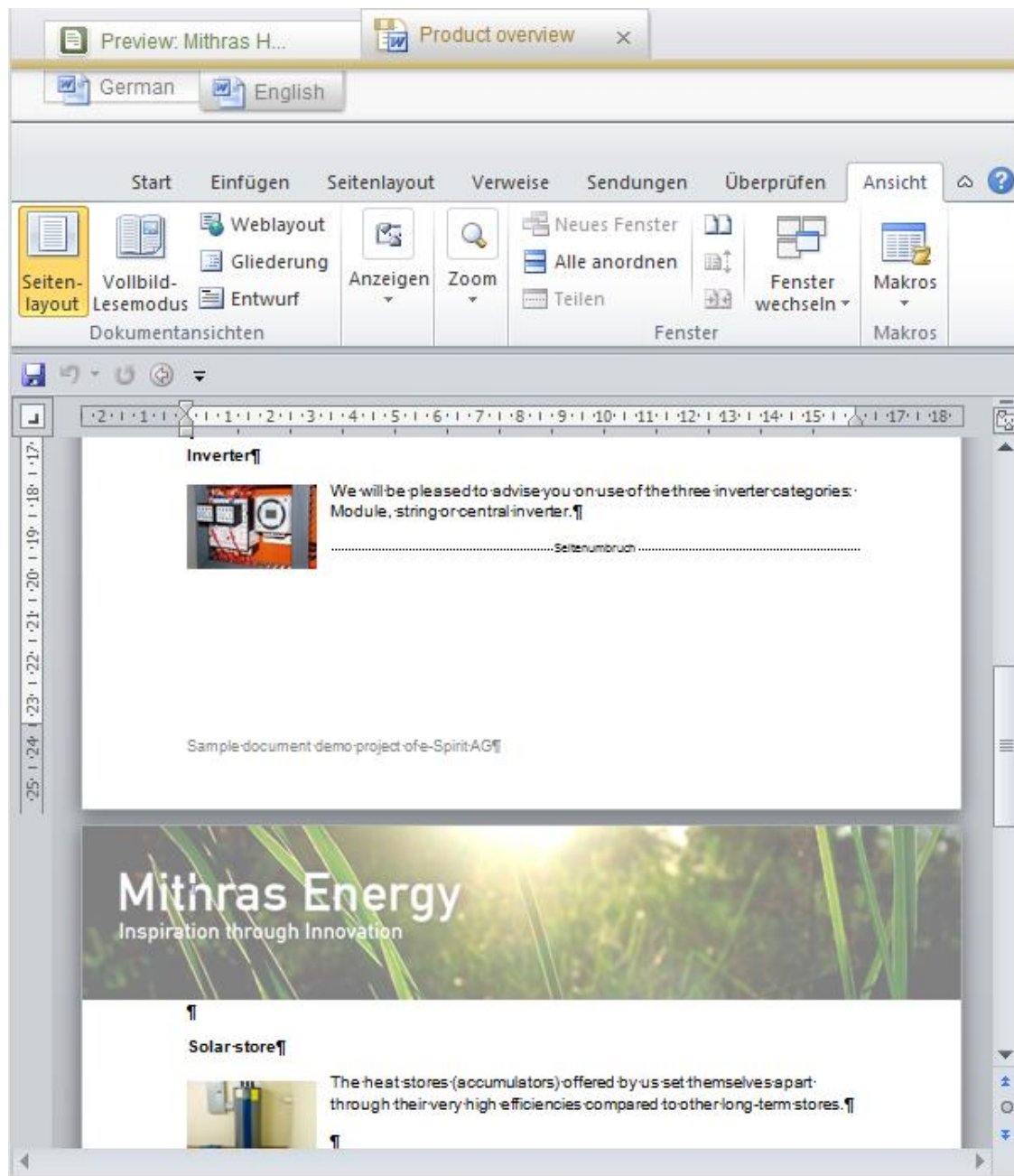



Figure 3-72: Displaying a Microsoft Word document in the FirstSpirit AppCenter



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

- the preview icon of the JavaClient tool bar
- CTRL + P
- the "Preview" entry of the context menu
- the new preview icon on the folder level of the Media Store
- switching to Edit mode


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.

Synchronization with the tree structure is always active. In other words, 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 tool tip 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 application area/Close other application areas:** 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.

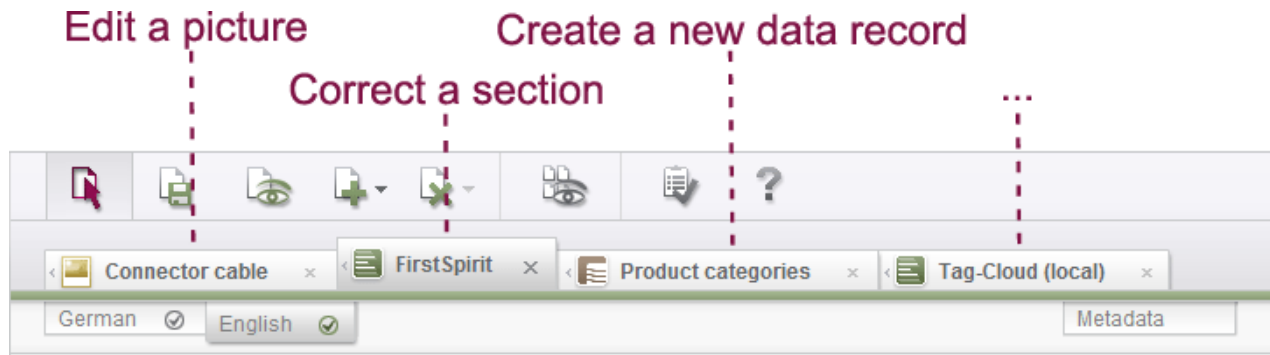




## 3.5 The editing area

### 3.5.1 Editing in several workspaces (horizontal tab navigation)

The editing area of the FirstSpirit JavaClient includes a horizontal tab navigation. In other words, 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 editor. This "horizontal tab navigation" enables users to very conveniently and clearly edit several workspaces. The order of tabs can be altered easily by drag-and-drop.



**Figure 3-73: Horizontal tab navigation**

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

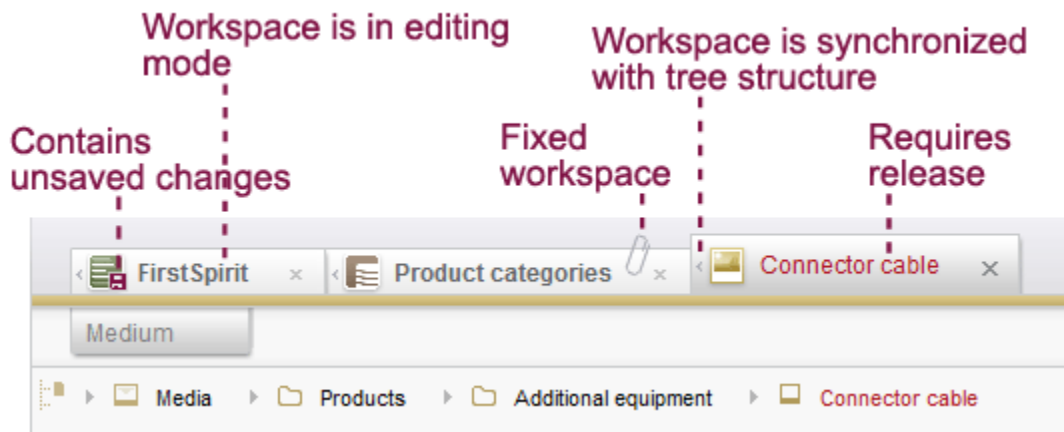
In this way the editor can set up their own personalized workspaces which ideally suit their tasks. The respective focus is retained when the tab is changed, so that the user is always displayed the last edited area in the respective tab (for example, a specific input form within a form).

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

- Ctrl + TAB: Select the workspace to the right of the active workspace.
- Ctrl + Shift + TAB: Select the workspace to the left of 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 labeled with a paper clip and are retained for all actions, until they are explicitly closed by the editor:





**Figure 3-74: Display of fixed and unfixed workspaces**

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

The **change state of an element** is denoted by three colors:

- 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 color coding for an element's change state is limited to these three colors and may be different than color coding in the tree structure.

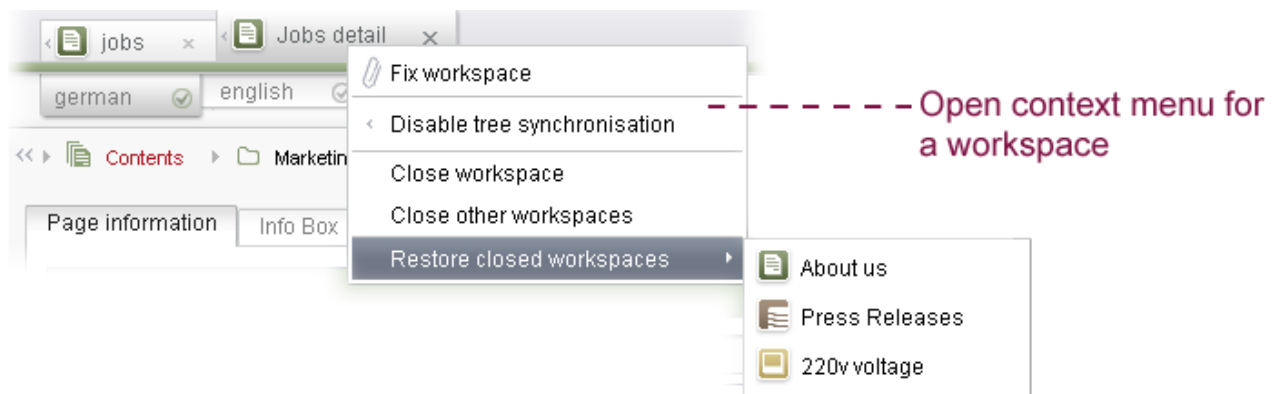
Apart from the change status (color of the lettering) the display also visually shows whether an element in an open workspace is locked to prevent editing (lettering is bold) or not and whether unsaved changes exist in a workspace (see Figure 3-74).

The individual workspaces can be moved within the tab area by means of drag-and-drop.

Apart from editing in individual workspaces, it is also possible to **drag-and-drop between the individual workspaces**. For example, media from the thumbnail view of a workspace can be copied into an image input component of another workspace by means of drag-and-drop (see section 11.4, page 310).

The "**Restore closed workspaces**" function, which is opened via the context menu on each tab of an open workspace, can be used to re-open a workspace that has already been closed (see also section 3.1.5.4, page 58).





**Figure 3-75: 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 section 3.1.5.4 page 58) as well as locally for each individual workspace using the context menu of the respective tab (see Figure 3-75).

Automatic synchronization of the active workspace in the right-hand area of the JavaClient with the tree display in the left-hand workspace (see also section 3.1.5.4, page 58) can be activated or deactivated using the context menu of the respective tab ("**Enable/Disable tree synchronization**" see Figure 3-75). Synchronization 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.

### 3.5.2 Breadcrumb navigation

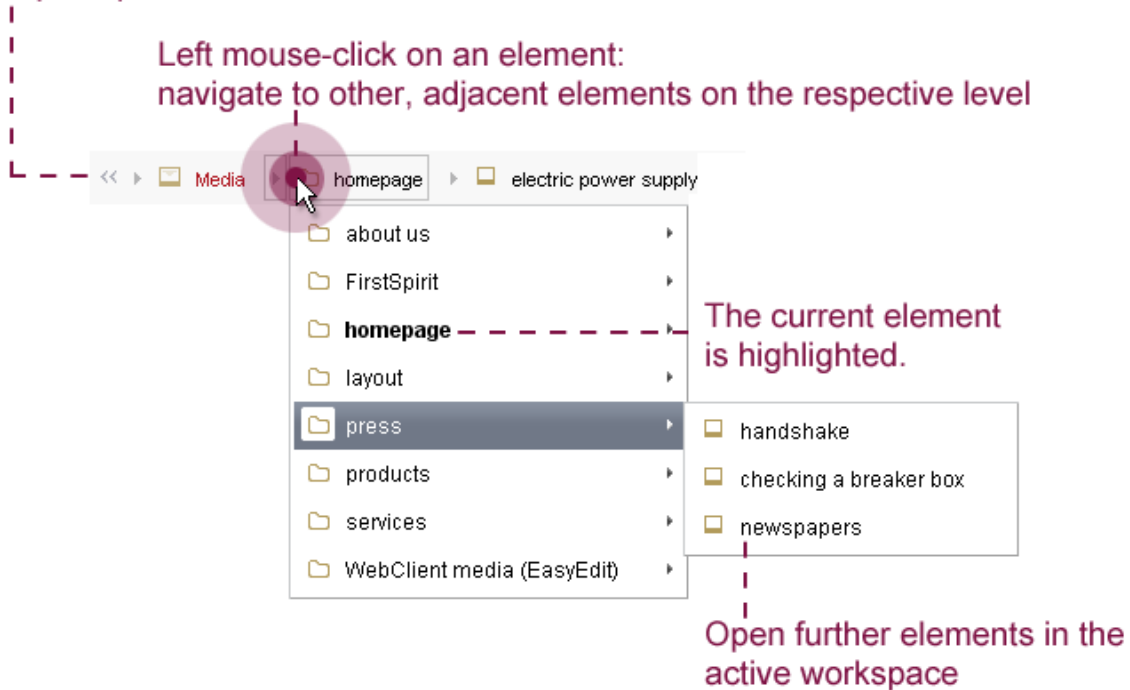
The path from the store root up to the current element is displayed above the form area in the form of "breadcrumb navigation." This display is particularly helpful if synchronization of the workspace with the tree is not enabled (see section 3.1.5.4 page 58 and section 3.5.1, page 120).

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.



**Breadcrumb navigation:**

Complete path from the store root to the element



**Figure 3-76: Navigation using the path elements ("breadcrumb navigation")**

In the same way as 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.

Like the tab lettering of the workspaces, the color coding for visualization of an object's change status (black, red, blue) is also used within the breadcrumb navigation (see section 3.5.1, page 120).

**3.5.3 Enhanced workspace-view ("TabView")**

Depending on the settings in the "View"/"Workspaces" menu and the individual method of working, you can end up with a large number of workspace tabs open in the JavaClient. The right, left, down and up arrow icons can be used to navigate through the open workspaces or select the desired workspace from a list view. To simplify operation of the workspaces, they can now be displayed in a clearer view and controlled comfortably, depending on the project developer's/administrator's settings.




The "TabView" icon in the tool bar of the JavaClient can be used to display an overview of



the workspaces in the AppCenter area.

### 3.6 The project entry page



The project entry page is accessed via the  icon. It provides a personalized overview of the bookmarks, open tasks and possible actions, with the relevant icons and in the color of the respective store.

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 colors: 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 editor 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 section 11.1 page 305.)

### 3.7 The status bar

The following information is displayed in the status bar:

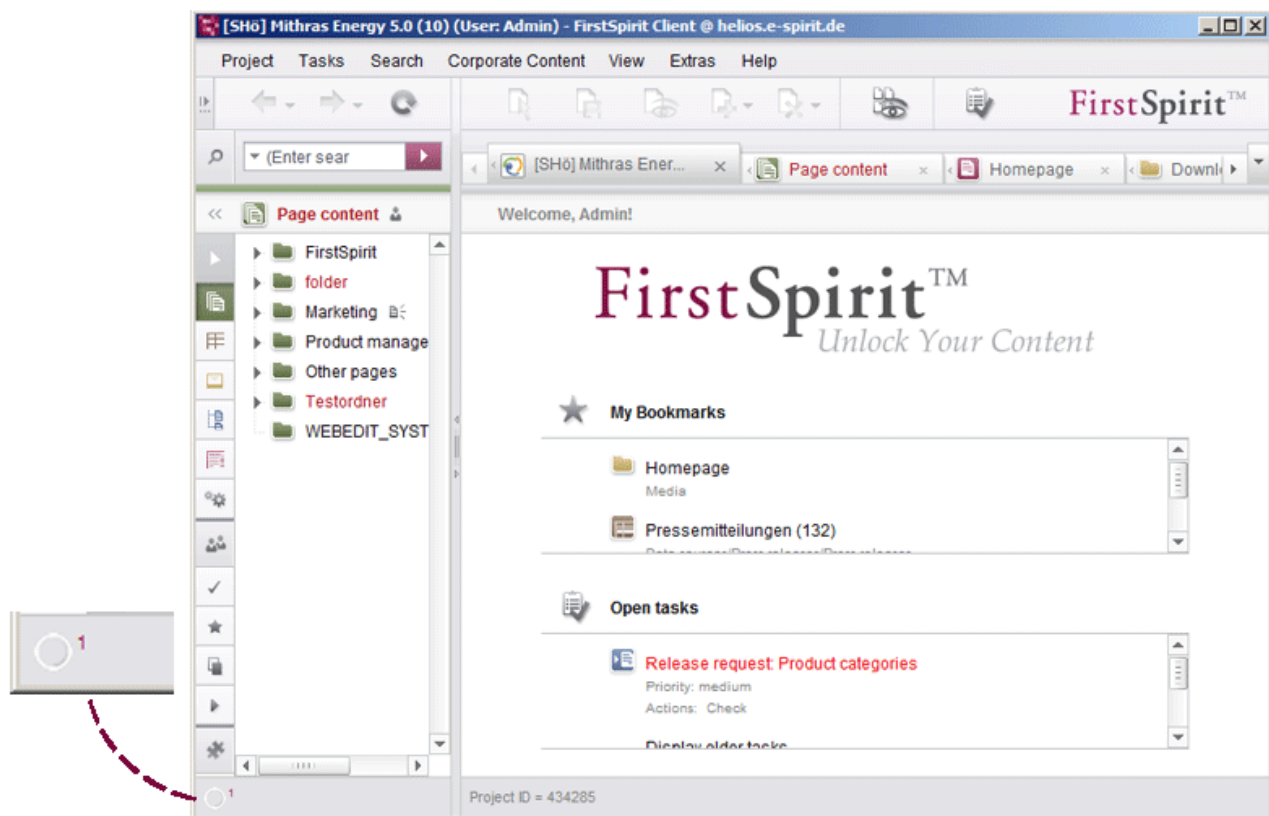
- Error messages from the system: Any errors that occur are briefly shown here in a loading icon and indicated by a small exclamation mark. Click the icon to open an information dialog with further details (see section 3.8, page 125).
- Status (released | not released | in workflow): a dropdown menu containing the next action/s which can be executed for the respective object can be opened with one click (see section 12, page 390)
- Version (date and time of the last editing): the version history of the respective object can be opened with one click (see section 11.10.2, page 371)
- Last editor
- In the case of pages and sections in the Page Store, the page or section template used is shown at the right-hand side. You can jump directly to this template in the Template Store by clicking on the template's name.



### 3.8 Error display

FirstSpirit provides a special infrastructure for collecting errors and exceptions. For this purpose, a loading icon is displayed in the bottom left area of the JavaClient which indicates continuously the data transfer while the doing editorial work.

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:



**Figure 3-77: Displaying an error message**

Click on this icon to get further information about the error that occurred. 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 stack trace 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.



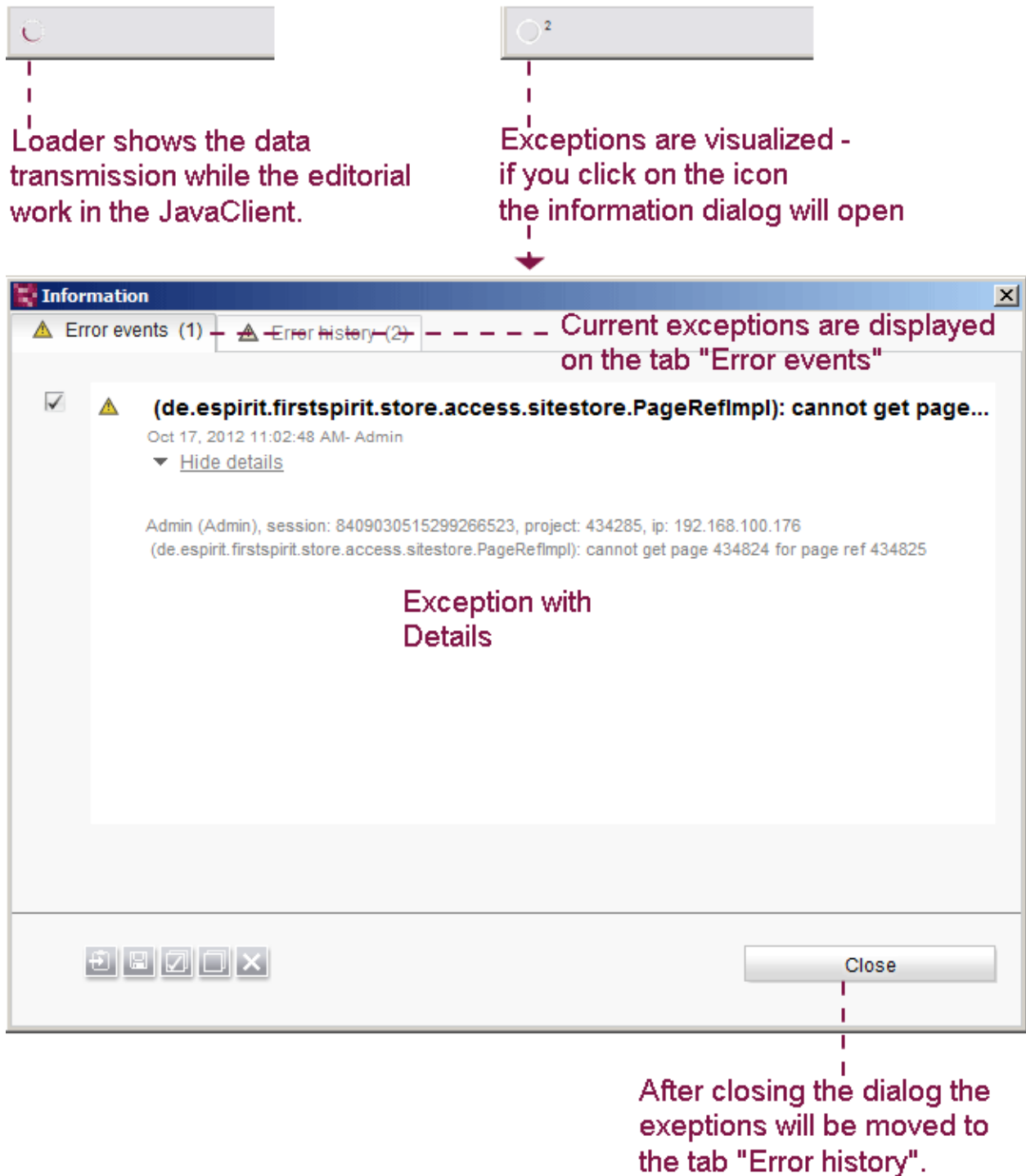


Figure 3-78: Central collection of errors in the FirstSpirit JavaClient

One or more error messages can be selected in the information dialog and copied to the clipboard and/or subsumed in an error 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, see section 4.3.11 page 142)



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.



Mark selected elements as seen: This icon is used to move the selected elements onto the "Error History" tab.



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  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.









## 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 an image 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 objects 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/data sources
-  Section references (see section 11.4.3, page 311 for how to create section references by drag-and-drop)

### 4.1 General context menus in the Page Store

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

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 "gray".
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 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 grayed out. 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.

### 4.1.1 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:

#### 4.1.1.1 New: Root node, folder and page

**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.

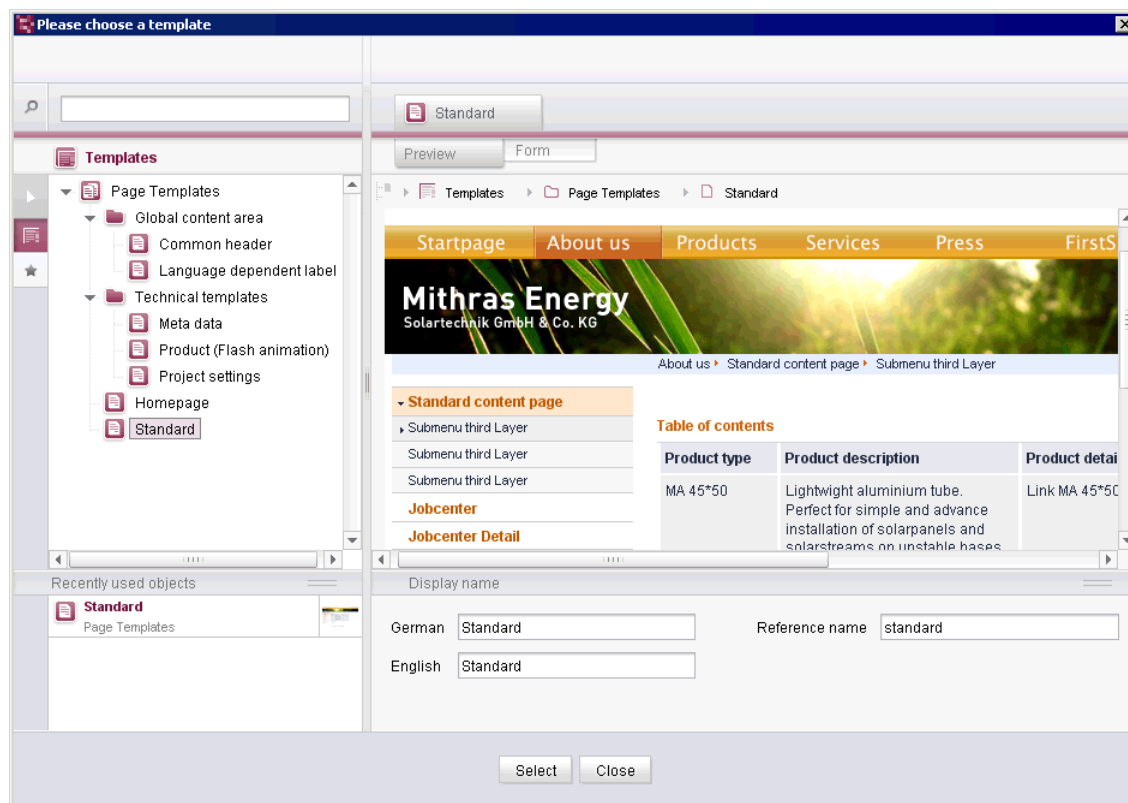


Figure 4-1: Insert new page – Choose template



You only have to select the required page template from the tree structure and confirm your selection by clicking **OK**.

Below the tree structure there is a list of the most recently used page templates of the respective user; this list is intended to make it easier to find frequently used templates.

In another window, you have to assign a name to the new page. Then the page appears in the selected position in the tree structure.

**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.

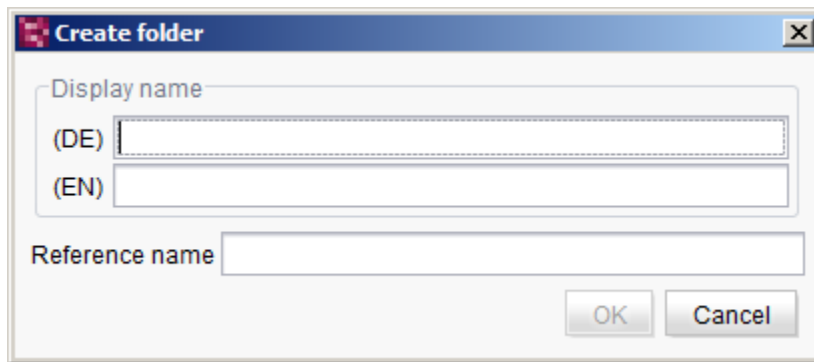


Figure 4-2: New – Create folder


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 "View – Preferred Display Language" menu (see section 3.1.5.2, page 55). 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.

If this function is called on a page, the new folder is always inserted beneath the folder containing the page.



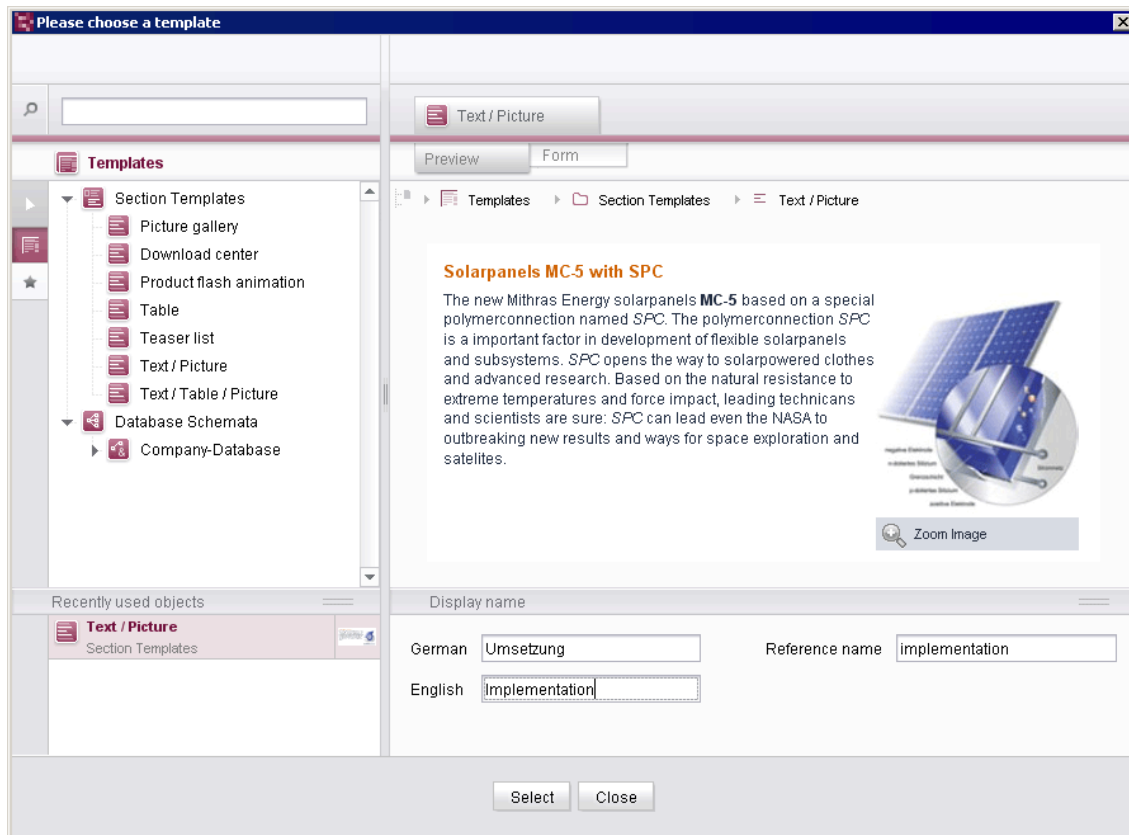
*The Reference Name field is only displayed if this setting was configured accordingly by the project administrator.*



 The project administrator can define rules to automatically convert special characters in reference names 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 name"). Special characters for which the administrator has not defined a replacement rule cannot be entered in the "Reference name" field.

#### 4.1.1.2 New: Content Area, Section and Content

**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 4-3: New – Choose 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 Editing 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.

#### 4.1.2 Lock/Unlock (Editing mode On/Off)

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

Opening this function again deactivates Editing mode, i.e. View mode is reactivated.

#### 4.1.3 Reset Changes

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

#### 4.1.4 Cut

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



#### 4.1.5 Copy

This function is used to generate a copy of the current object and store it in the clipboard. This copy can be inserted in another position in the tree structure.

#### 4.1.6 Paste

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

#### 4.1.7 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.



*When renaming sections, you can change both the display name and the reference name for the section. As no two of the sections underneath a page may have the same name, the "Reference name" field label is shown in red in the event of a conflict and the OK button is disabled so it is not possible to save the name.*

#### 4.1.8 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, for example for deleting objects, can be tied to this function (see section 3.2.9.7, page 87). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*



#### 4.1.9 Display in current/new workspace

These functions allow you to choose the number of open tabs in the editing area of the FirstSpirit Client.

**Display in current workspace** opens the selected object in the central part of the screen on the active tab. **Display in new workspace** opens the selected object in a new tab.

#### 4.1.10 Version history

A window opens in which all the old versions of the current object are listed. For precise documentation of the version history in the Page Store see section 11.10.3, page 380.

#### 4.1.11 Workflow

If no workflow is 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 or state under this menu item.

Detailed documentation of workflows is given in section 12 page 390.

#### 4.1.12 Execute Script

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

### 4.2 Special context menus in the Page Store

#### 4.2.1 CorporateContent

The "CorporateContent" context menu provides dedicated functions for editing packages directly on the objects in the tree structure. The context menu is divided into five sub-menu items: "Add to package," "Remove from package," "Undo package relation," "Change state" and "Rebind original."



All the functions are documented in detail in the *FirstSpirit CorporateContent* module documentation.

#### 4.2.2 Add to Content transport feature (at folder and page level)

This license-dependent function allows you to conveniently reuse project content between projects.

A detailed description of this is given in the *FirstSpirit CorporateContent* module documentation.

#### 4.2.3 Display preview (at page level)

If integrated preview for content is not enabled (see section 3.1.5.5, page 59), you can use this function to check how the page that was just edited will look on the website. Opening the function generates a preview of the page. This function can be carried out separately for all project languages as well as for the current and the released state of the page /content area.



The button  in the tool bar is also available for this function. (See section 3.2.7 page 81.)

#### 4.2.4 Show preview errors (at page level)

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

This function can be carried out separately for all project languages as well as for the current and the released state of the page /content area.

#### 4.2.5 Preview (release) (at page level)

Opening the function generates a preview of the page after the last release. No changes made since then are displayed. This function can be carried out separately for all project languages as well as for the current and the released state of the page /content area.





#### 4.2.6 Preview errors (release) (at page level)

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

This function can be carried out separately for all project languages as well as for the current and the released state of the page /content area.

#### 4.2.7 Export (at folder and page level)

This function is available to administrators only.

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 to 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 puts great strain on 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.*

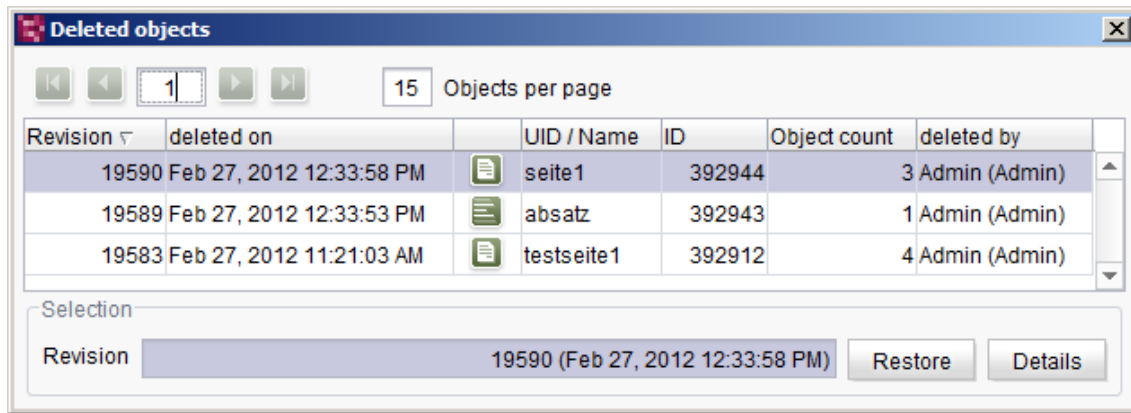
#### 4.2.8 Import (at root, folder and page level)

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.

#### 4.2.9 Restore deleted objects (at root, page, content area and 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 you click it, a window opens with the deleted objects.





**Figure 4-4: Deleted objects**

Only the deleted objects located beneath the selected object are ever displayed.

The following information is given for each object:

**Revision:** Version number of the deleted object.

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

**Type of object:** Object icon indicating the type of object that was deleted.

**Name:** The reference name of the deleted object.

**ID:** The unique ID number 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 by:** Name of the user who deleted the object.

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

Restoration options and the position of the object to be restored can be determined in two further dialogs.

#### 4.2.10 Change position (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 moving one position **up** or **down** or, helpful for pages with many sections, directly at the **first** or **last** position.

### 4.3 Functions under the Extras context menu of the Page Store

#### 4.3.1 Release (at root, folder and page level)

This function is only available for administrators.

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 section 12.4 page 399.

#### 4.3.2 Delete metadata (at root, folder and page level)

This function is only available for administrators.

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

#### 4.3.3 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 section 13.1.2, page 409.

#### 4.3.4 Reset write protection

This function is available to administrators only.

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

#### 4.3.5 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. The name of the template used is also displayed at the bottom right in the status bar (see section 3.7, page 124). You can jump to the corresponding template in



the Template Store with one click.

#### 4.3.6 Select another template (at page and section level)

This function is available to administrators only.

This function can be used to select a different template for individual pages or sections. After clicking, a window opens containing the available page templates or section templates.



*If existing sections of a page cannot be copied to the new page template, the action is canceled with the message "The selected template cannot be used."*

#### 4.3.7 Jump to source section (at section reference level)

Using this function, the referenced section is displayed in the tree structure and can be edited if required (and if authorized) (see also section 11.4.3, page 311).


#### 4.3.8 Edit validity period (at section level)


This function is only available for administrators.

This function can be used to specify in which period a section should be presented on the website.

**Edit validity period**

Define lifespan

Validation start date  

Validation end date  

or  Unlimited validity

OK Cancel

Figure 4-5: Edit validity period



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

**Validation start date:** Click the **calendar icon** after the line to open a new window in which the validation start date can be set to the nearest minute:

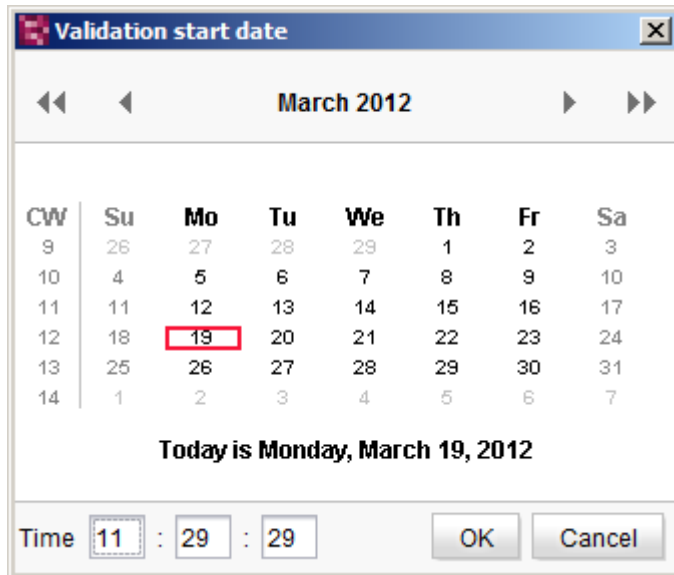


Figure 4-6: Define lifespan

**Validation end date:** The end of the validity can then be set in the same way by clicking the calendar icon after 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.



Figure 4-7: Status bar with defined validity period

The validity period is displayed in gray if the current date is within the specified period. The validity period is displayed in red if the current date is outside the specified period.



### 4.3.9 Show usages (at page level)

This function can be used to determine where in the Site 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.

### 4.3.10 Change reference name (at root, folder and page level)

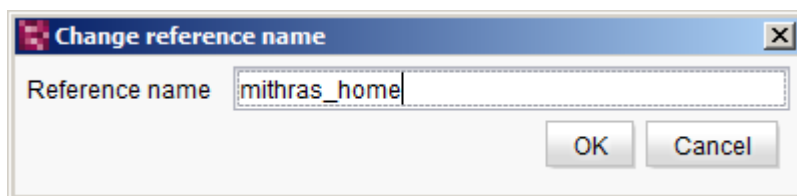
This function is only available for administrators.

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 "Display reference names in tree" checkbox and the menu item View – Preferred display language have been activated (See section 3.1.5.2, page 55).*

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 name" 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. an image 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 4-8: Change reference name**

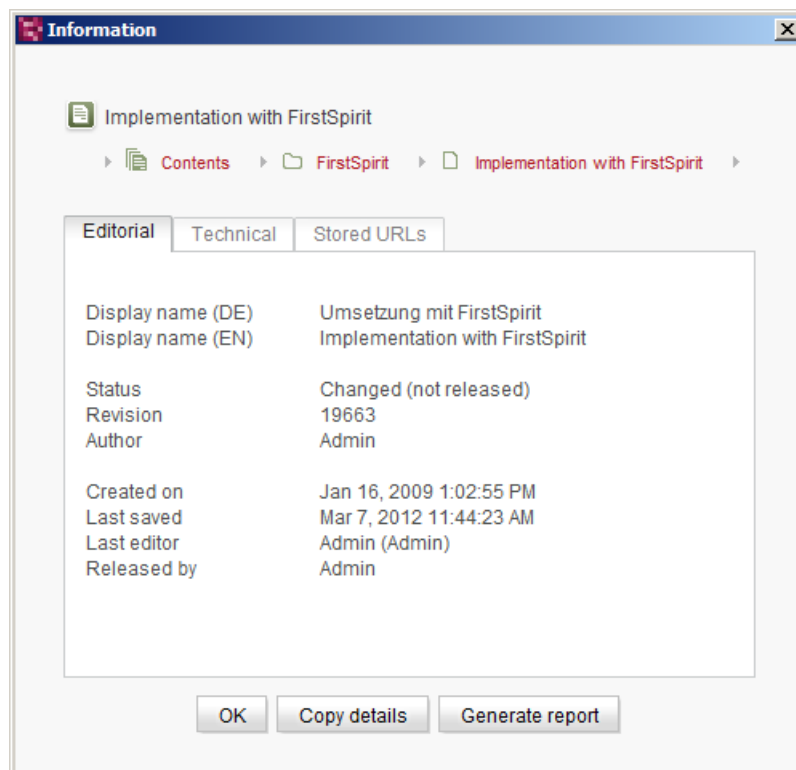
If the reference name is changed despite an existing reference, it may be necessary to restore the reference manually using a new selection.



Depending on the project administrator's settings, the menu entry may be grayed out regardless of the "Change" permission; in this case the reference name cannot be changed.

#### 4.3.11 Display properties

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. This function can also be called using the keyboard shortcut "ALT + P". The information can vary, depending on the object type.



**Figure 4-9: Properties of a page – Editorial**

This path allows you to navigate to other objects in the tree structure. This makes it possible to display the properties of other objects without closing the window.

**OK:** Click this button to close the dialog.

**Copy details:** Use this button to copy all the dialog's information onto the clipboard.

**Generate report:** Use this button to output the information as an HTML page in the form of a system report (see section 3.8, page 125). In addition, an additional comment can be entered, for example, an error description.



#### 4.3.11.1 Editorial tab

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

**Display name:** The language-dependent display names of the object are displayed.

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

**Last saved:** 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

#### 4.3.11.2 Technical tab

The technically relevant properties of an object are displayed on 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)



+ Q.

*The path information can also be requested using the keyboard shortcut Ctrl + Shift*

**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**" link is displayed, with which it is possible to switch directly to the template on which the object is based.

#### 4.3.11.3 Stored URLs tab (at page level)

The URLs stored for the current page are displayed on this tab. They can be reset to page references via the context menu entry "Extras – Reset stored URLs". (For more details of the URL settings, see section 9.3, page 260.)

#### 4.3.12 Cancel editing (at page and section level)

Using this function the Editing mode can be ended without accepting changes which have been not saved yet. The status of pages that have already been released is unaffected by this action.

#### 4.3.13 Display dependencies (at page and section level)

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

For a detailed documentation of the reference graph please refer to section 11.11 page 386.



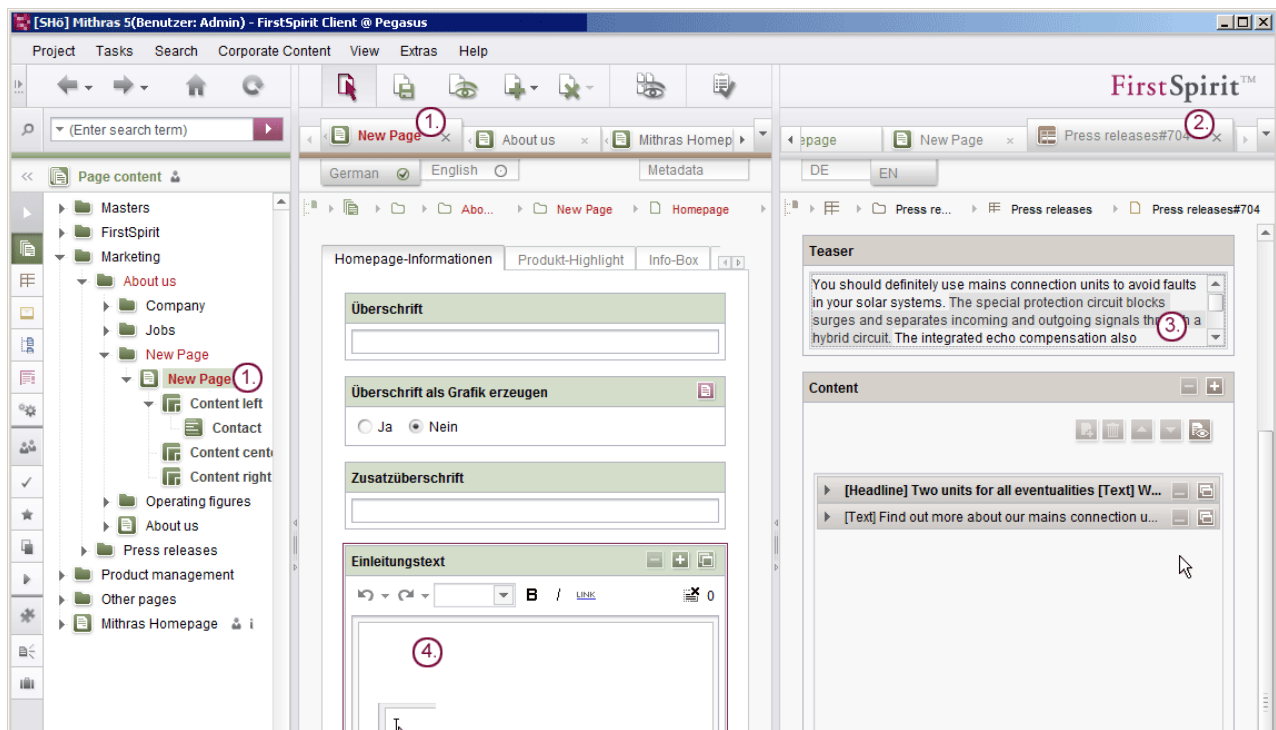
## 4.4 Functions under the Plug-ins context menu of the Page Store

### 4.4.1 Open FormView AppTab (at page and section level)

Depending on the project developer's settings, this function can be used to call up a comparison view for the selected objects in the AppCenter area.

This makes it possible to permanently display sections, pages and datasets in their form view in the preview in order to compare the content of their input forms with other input forms (e.g. when revising pages, sections or datasets) or in order to copy content from these input forms to input forms in the workspace.

A new tab opens in the AppCenter area showing the input forms for the selected object. No content can be edited on this tab.




**Figure 4-10: Opening a dataset in the integrated preview**

In the example shown here, a page and its available input forms were initially switched to Editing mode in the workspace (1.). The content of a dataset can be used to input content. The dataset is displayed via the script on the dataset in the integrated preview (2.). You can use copy & paste (3.) to copy content from the input forms available there. Content is added to the input forms in



the workspace (4.). It would also be conceivable to display the content of an element in two different languages side by side.

The tab opened here remains until it is closed by clicking the X on the right-hand edge of the tab or selecting the context menu entry "Close preview" or "Close other previews".

 *The content of multiple workspaces can also be compared using the enhanced workspace view ("TabView") (see section 3.5.3, page 123).*

## 4.5 Settings at page level

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

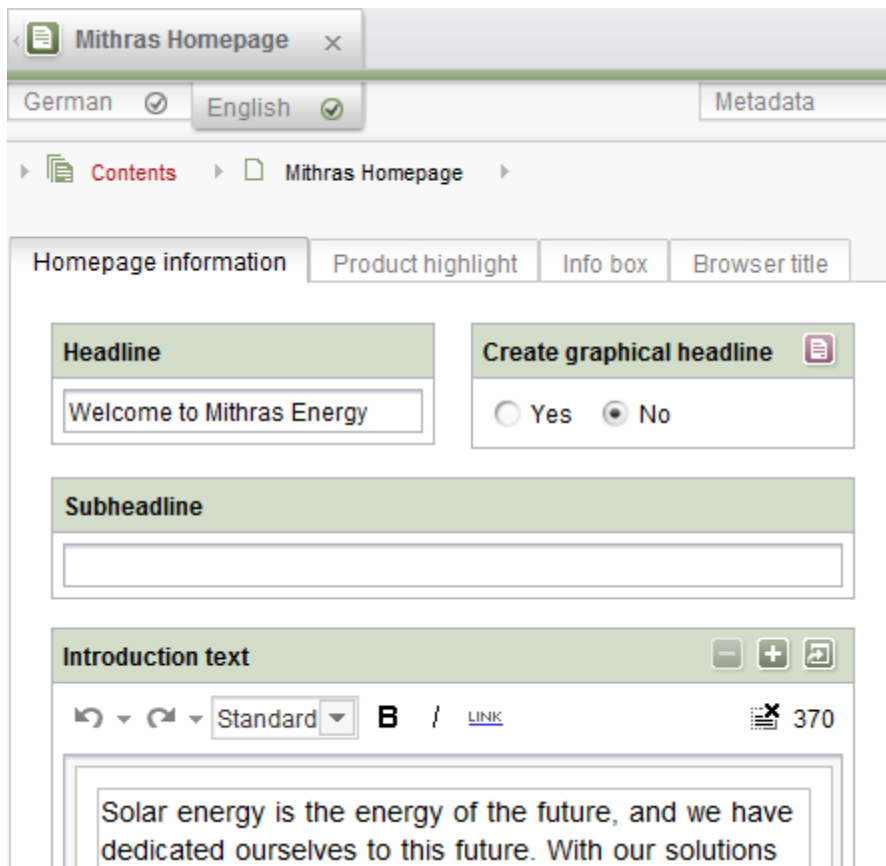


Figure 4-11: Page view

**Path:** The path shows the user's current position within the tree structure, e.g. in which folder, on



which page, in which section, in which medium, etc. the user is currently located.

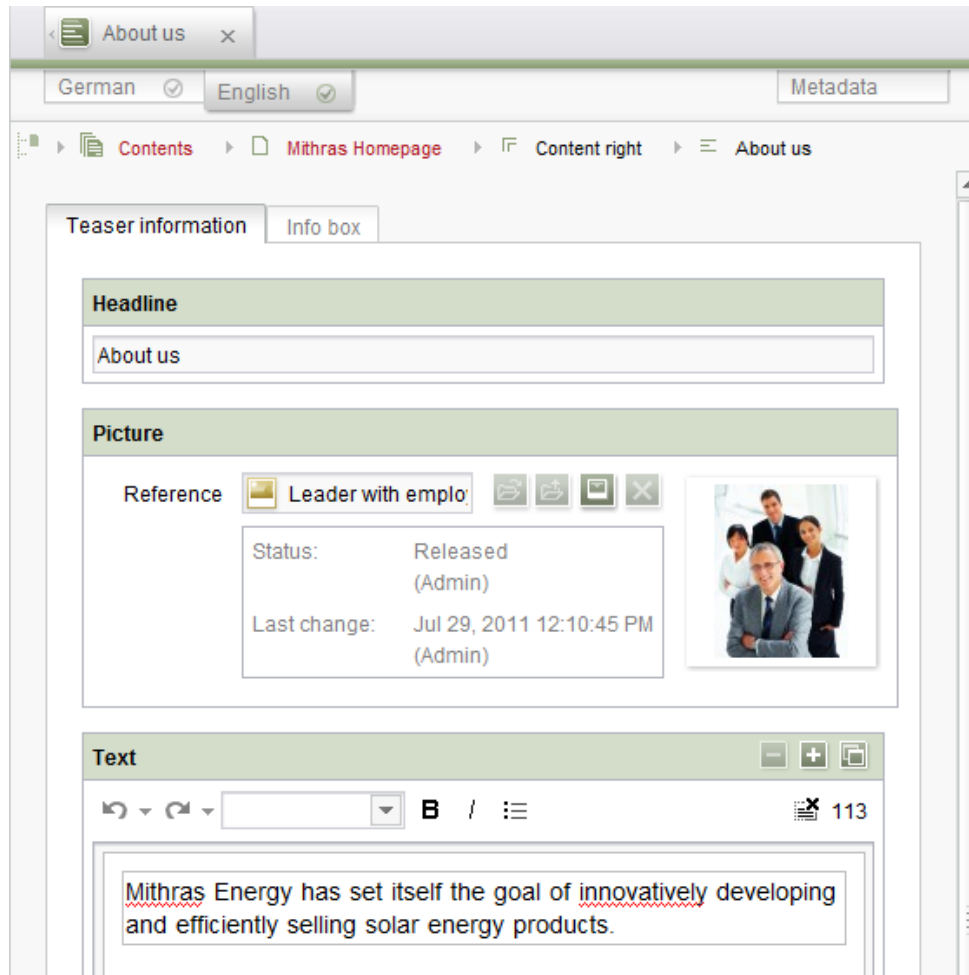
**Page is completely translated to this language:** The check marks on the language tabs (in multilingual projects) indicate that the page is completely translated to this language. New pages are created with a check mark for all available languages by default. This option can be deactivated if not all the content is available in the relevant language. In this case, the page is not taken into account in the corresponding language when the project is generated, depending on the project settings. Although it is also possible to switch between language tabs in view mode, the checkbox can only be activated or deactivated in Editing mode for the page.

**Input components:** This area may contain various input components that the template developer has earmarked for this page. Detailed documentation on the available input components is given in section 10, page 273.



## 4.6 Settings at section level

After a new section has been inserted in the content area of a page, Editing mode is automatically activated for this page. The section can now be edited.



**Figure 4-12: Section view**

**Path:** The path shows the user's current position within the tree structure, e.g. in which folder, on which page, in which section, in which medium, etc. the user is currently located.

**Include this section in the output:** The check marks on the language tabs (in multilingual projects) indicate that the section in the corresponding language will be taken into account during the next generation. This option can be deactivated if not all the content is available in the relevant language. In this case, the section is not taken into account in the corresponding language when the project is generated.

**Input components:** This area may contain various input components that the template



developer has earmarked for this section. Detailed documentation on the available input components is given in section 10, page 273.

#### 4.6.1 Section references

Section references can be used to reuse the content of a section on other pages. A section reference can be generated by drag-and-drop (see section 11.4.3 page 311).

A section reference cannot be edited, only the source section can be edited. To identify the source section, use the "Jump to source section" function in the context menu (section 4.3.7, page 139).



*It is not possible to create section references from sections that are based on table templates.*

If the source section contains **metadata**, the metadata is copied to the section reference. No metadata can be defined. For more information about the evaluation of metadata see section 11.3 page 309.






## 5 Content Store of the JavaClient

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

The tables of a database can be integrated and maintained in the Content Store in the form of tables. They are then placed on a page as what are known as data sources via the Page and Site Stores.

**The following elements can be created within the Content Store:**

-  Folders for structuring the data sources.
-  Data source
-  Filtered data source (see sections 5.3.4 and 5.5.2.2, pages 160 and 166)

### 5.1 General context menus in the Content Store

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

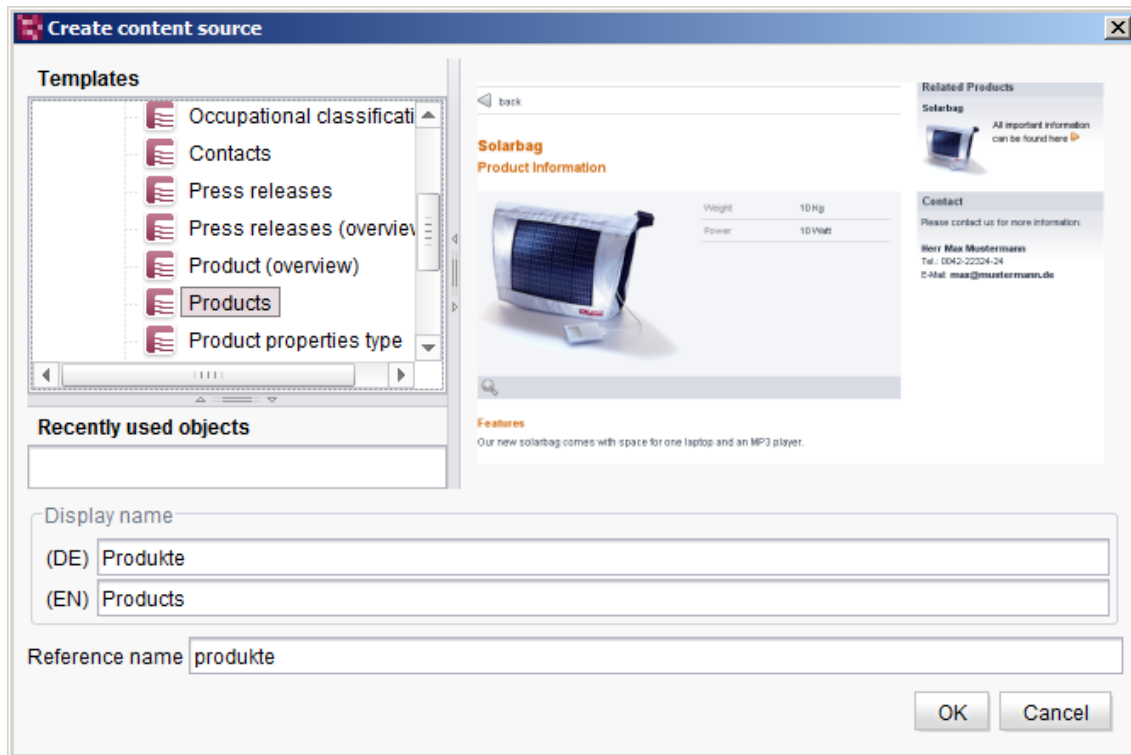
For general information on handling context menus see section 4.1 General context menus in the Page Store page 128.

#### 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:

**New – Create content source:** 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 5-1: New – Create table**

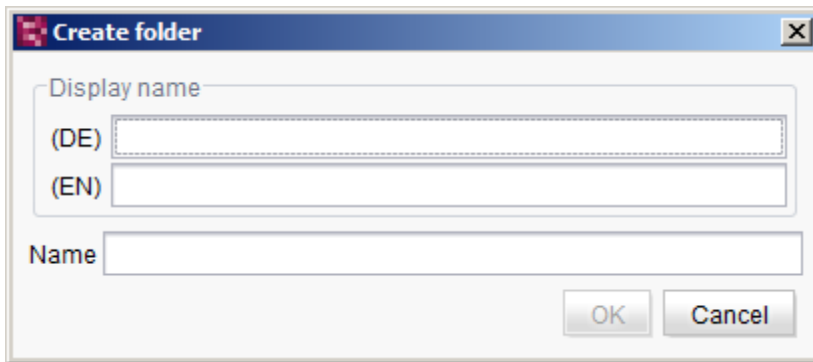
You only have to select the required table template from the tree structure and confirm your selection with **OK**. Below the tree structure there is a list of the most recently used templates of the respective user; this is intended to make it easier to find frequently used templates.

The display name of the selected table template is automatically adopted for new data sources 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 for a data source must be unique, this value is uniquely assigned by appending numbering.

**New – Create folder:** It is advisable to also file the individual data sources in folders in the Content Store 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 5-2: 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 "View – Preferred Display Language" menu (see section 3.1.5.2 page 55). 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.

If this function is called on a data source, the new folder is always inserted beneath the folder containing the data source.



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



*The project administrator can define rules to automatically convert special characters in reference names 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 name"). Special characters for which the administrator has not defined a replacement rule cannot be entered in the "Reference name" field.*



### 5.1.2 Editing on/off

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

Unlike in other stores, a data source will never be locked when switched to Editing mode so that different datasets 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 dataset simultaneously, what is known as the "optimistic lock method" is applied (see section 5.5.4, page 172).



*The project administrator has the option of activating exclusive editing of datasets instead of the optimistic lock method. In this case, a dataset can be switched to Editing mode so that no other editors can make any changes to the dataset.*

Opening this function again deactivates Editing mode for the data source and the dataset which is currently in Editing mode. Changes are saved and the view mode becomes active.



*If a data source is to be refreshed (via F5 or the respective icon in the tool bar (see section 3.2.4, page 78)) for example to show modifications made by other users (e.g. creating new datasets), 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 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 clipboard. This copy can be inserted in another position in the tree structure.





*At the level of a dataset, "Copy" copies the view of a database table only, not the selected dataset.*

### 5.1.5 Paste

This function is used to insert the content of the clipboard in the current position of the tree structure. This function is only active if there is data in the 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

This function can be used to delete an empty folder from the tree structure of the Content Store. Accidental deletion is prevented by a confirmation prompt.



*A workflow, for example for deleting objects, can be tied to this function (see section 3.2.9.7, page 87). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

### 5.1.8 Display in current/new workspace

These functions allow you to choose the number of open tabs in the editing area of the FirstSpirit Client.

**Display in current workspace** opens the selected object in the central part of the screen on the active tab. **Display in new workspace** opens the selected object in a new tab.



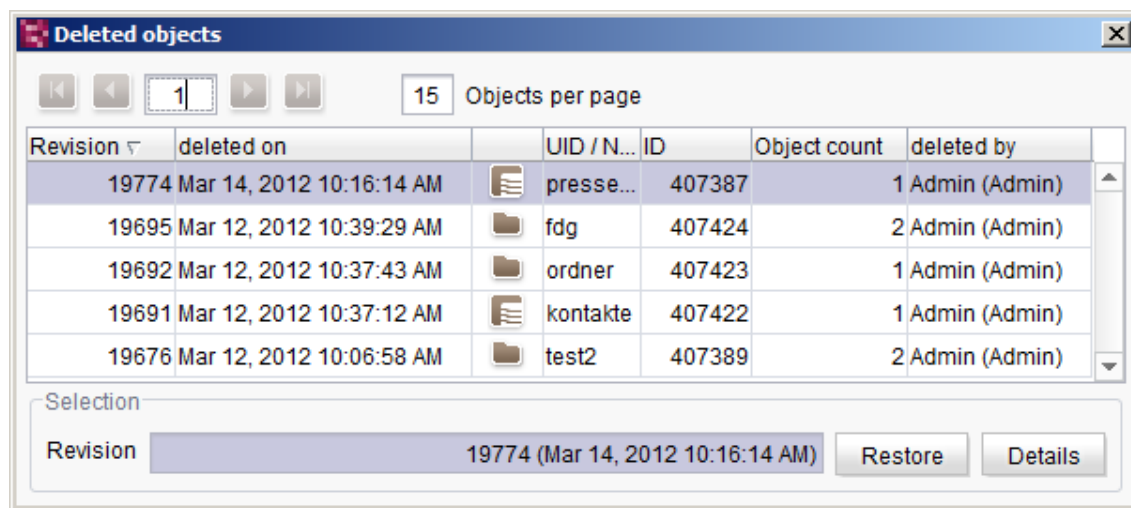
## 5.2 Special context menus in the Content Store

Some of the functions from the special context menu can also be called via the context menu in the workspace at the dataset level.

### 5.2.1 Restore deleted objects

#### 5.2.1.1 At folder level – Restore deleted objects

If a folder or a data source has been accidentally deleted from the tree structure this function can be used to restore the deleted object. After you click it, a window opens with the deleted objects.



**Figure 5-3: Deleted objects – folder level**

Only the deleted objects located beneath the selected folder are ever displayed.

The following information is given for each object:

**Revision:** Version number of the deleted object.

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

**Type of object:** Object icon indicating the type of object that was deleted.

**Name:** The reference name of the deleted object.

**ID:** The unique ID number 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 by:** Name of the user who deleted the object.

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

Restoration options and the position of the object to be restored can be determined in two further dialogs.

Wherever possible, restored objects at the page or content area level 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 for example 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.

#### 5.2.1.2 At data source level – Restore deleted objects

If a dataset has been deleted from the table by mistake the deleted dataset can be restored using this function. After you click it, a window opens with the deleted datasets.

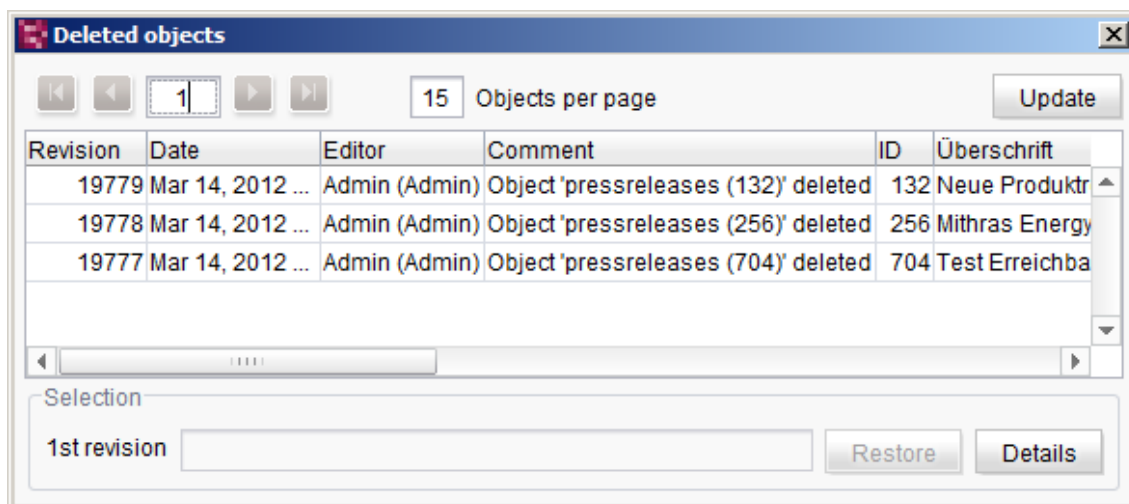


Figure 5-4: Deleted objects – data source



The following information is given for each dataset:

**Revision:** Version number of the deleted object

**Date:** Date and time when the dataset was deleted

**Editor:** Name of the user who deleted the dataset

**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 datasets may have been added in the meantime.

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

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

## 5.2.2 CorporateContent (at folder and data source level)

The "CorporateContent" context menu provides dedicated functions for editing packages directly on the objects in the tree structure. The context menu is divided into five sub-menu items: "Add to package," "Remove from package," "Undo package relation," "Change state" and "Rebind original."

All the functions are documented in detail in the *FirstSpirit CorporateContent* module documentation.



### 5.2.3 Add to Content transport feature

This license-dependent function allows you to conveniently reuse project content between projects.

A detailed description of this is given in the *FirstSpirit CorporateContent* module documentation.

### 5.2.4 Add all displayed datasets to feature (data source)

This license-dependent function allows you to conveniently reuse project content between projects.

A detailed description of this is given in the *FirstSpirit CorporateContent* module documentation.

### 5.2.5 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 Content Store see section 11.10.4 page 382.

### 5.2.6 Starting workflow

If no workflow is 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 or state under this menu item.

Detailed documentation of workflows is given in section 12 page 390.

### 5.2.7 Execute Script

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



### 5.2.8 Show usages (at dataset level)

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

### 5.2.9 New bookmark (at dataset level)

This function can be used to set bookmarks for individual datasets in the Content Store. For information about using bookmarks, see section 3.2.2, page 76.

### 5.2.10 Display dependencies (at dataset level)

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

For a detailed documentation of the reference graph please refer to section 11.11 page 386.

## 5.3 Functions under the Extras context menu of the Content Store

### 5.3.1 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 section 13.1.2, page 409.

### 5.3.2 Release displayed lines (data source)

This function is only available for administrators.

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

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





### 5.3.3 View Template (data source)

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

### 5.3.4 Set filter (data source)

This function is used to filter a data source for datasets and can also set a sort order. This makes it easier to work with data sources which may possibly contain many similar datasets.



To create filters, the user needs the "Create folders" permission (see section 13.1.4.6, page 423). To further edit filtered datasets (see section 5.5.2.2, page 166). On the other hand, "Change" permission is sufficient (see section 13.1.4.4, page 422).

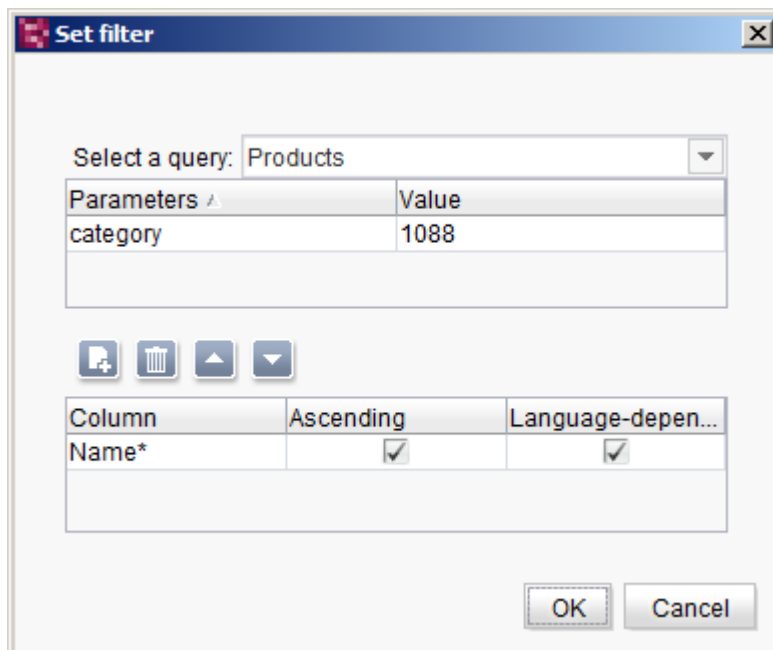


Figure 5-5: Filter datasets

At the top of the dialog

**Select a query:** Use this drop-down list to select from the Template Store a database query defined in advance by a project developer. This query is used for filtering. (See *FirstSpirit Manual*



for Developers)

After a query is selected, all the parameters specified 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.

#### At the bottom of the dialog

The sort order of the filtered datasets can be defined here. 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 ordering by column: click this icon to use a selection box to specify another table column as the sort order.





Remove sort order: click this icon to delete the activated sort order.



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

If several sort orders are specified, the top sort order in the list is used for the sort order first. All entries for which this first sort order is identical are then sorted by the second sort order, 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 editing area for the data source now only shows the filtered datasets 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 an icon for filtered data sources . Filters and sorting can be changed again by opening the **Set filter** function from the context menu on the respective node. You can restrict the search results further by selecting multiple filters.

### 5.3.5 Delete data source

This function is available to administrators only.

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



### 5.3.6 Reset write protection

This function is available to administrators only.

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

### 5.3.7 Change reference name (data source)

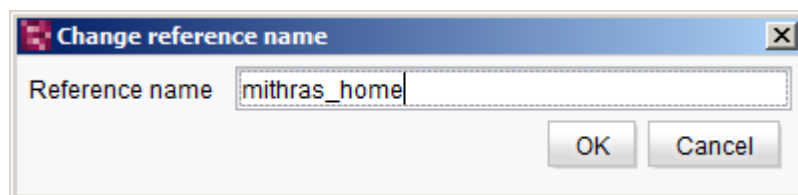
This function is only available for administrators.

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 "Display reference names in tree" checkbox and the menu item View – Preferred display language have been activated. (See section 3.1.5.2, page 55.)*

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 name" 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. an image 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 5-6: Change reference name**

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

Depending on the project administrator's settings, the menu entry may be grayed out regardless of the "Change" permission; in this case the reference name cannot be changed.



### 5.3.8 Display dependencies (data source)

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

For a detailed documentation of the reference graph please refer to section 11.11 page 386.

### 5.3.9 Display properties

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. This function can also be called using the keyboard shortcut "ALT + P". The information can vary, depending on the object type.

*For further information about this function see section 4.3.11, page 142.*

## 5.4 Functions under the Plug-ins context menu in the Content Store

### 5.4.1 Open FormView AppTab (dataset)


Depending on the project developer's settings, this function can be used to call up a comparison view for the selected dataset in the AppCenter area.

*For further information about this function see section 4.4.1, page 145.*

## 5.5 Tables in the Content Store

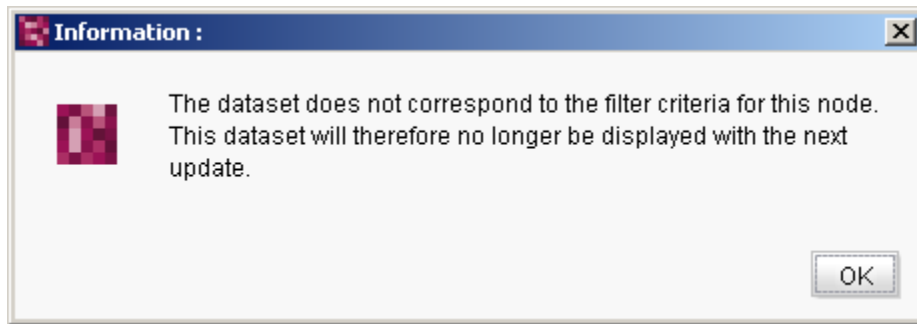
After a new table has been inserted in the Content Store, datasets can be entered in the right-hand side of the screen. If datasets already exist they can also be further edited here (see section 5.5.4 page 172).

### 5.5.1 Filtered data sources

Data sources with the  icon were permanently filtered using the "Extras" "Set filter" function (section 5.3.4, page 160).

If a new dataset is created in this type of data source, or an existing dataset is changed so that it no (longer) satisfies the defined filter criteria, the following message is output after the dataset is saved:





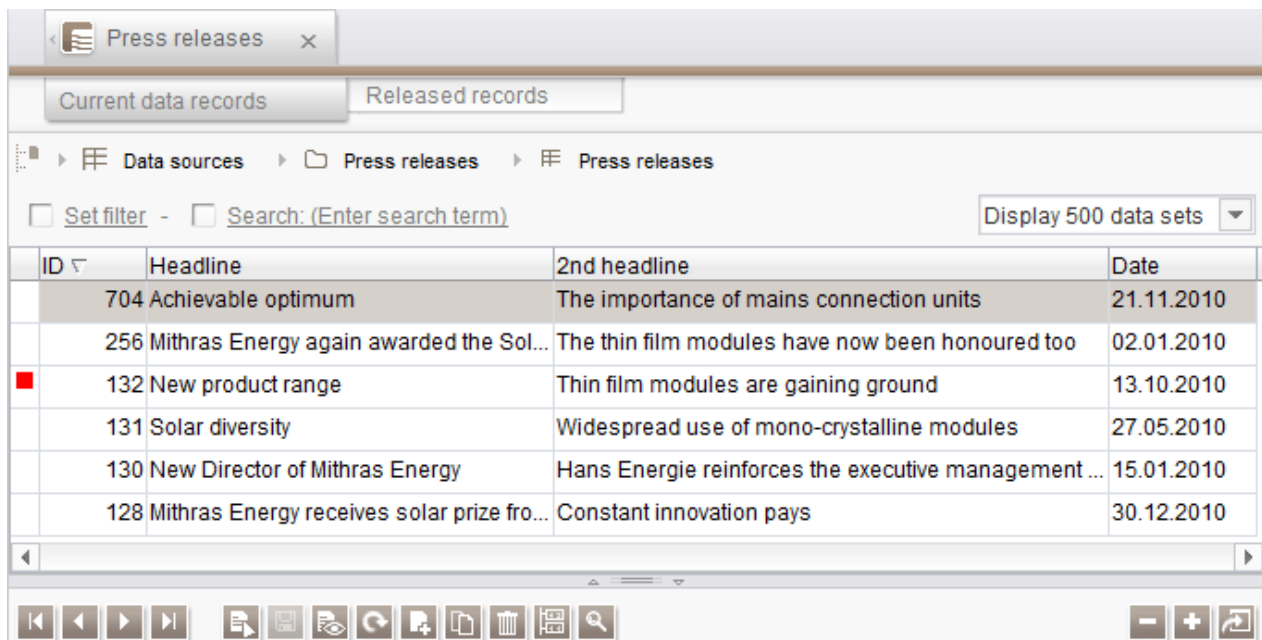
**Figure 5-7: Dataset does not satisfy filter criteria**

The dataset is no longer displayed following an update, e.g. using <F5> or a change to another node.

Otherwise, the same functions are available on filtered data sources as on data sources that have not been filtered (see the next section).

### 5.5.2 Data overview – Current data records

If a data source is selected in the tree structure, the overview of the current datasets is displayed:



**Figure 5-8: Table overview – Current datasets**



All the available datasets are listed in the table view. They are displayed on two tabs, separated according to current and released (see section 5.5.5, page 175) datasets.

Click a column heading to sort the lists by the values in the respective column. Current datasets can be sorted by their modification and release status via the first column (modified and unreleased datasets are displayed at the top or bottom), the datasets' ID is used as a second sort criterion.

As all the columns of a dataset 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. In addition, you can change the column width by dragging the right-hand edge of the column label.



5.5.2.1 Multi-line display

Datasets can be displayed in multiple lines, depending on the template developer's settings:

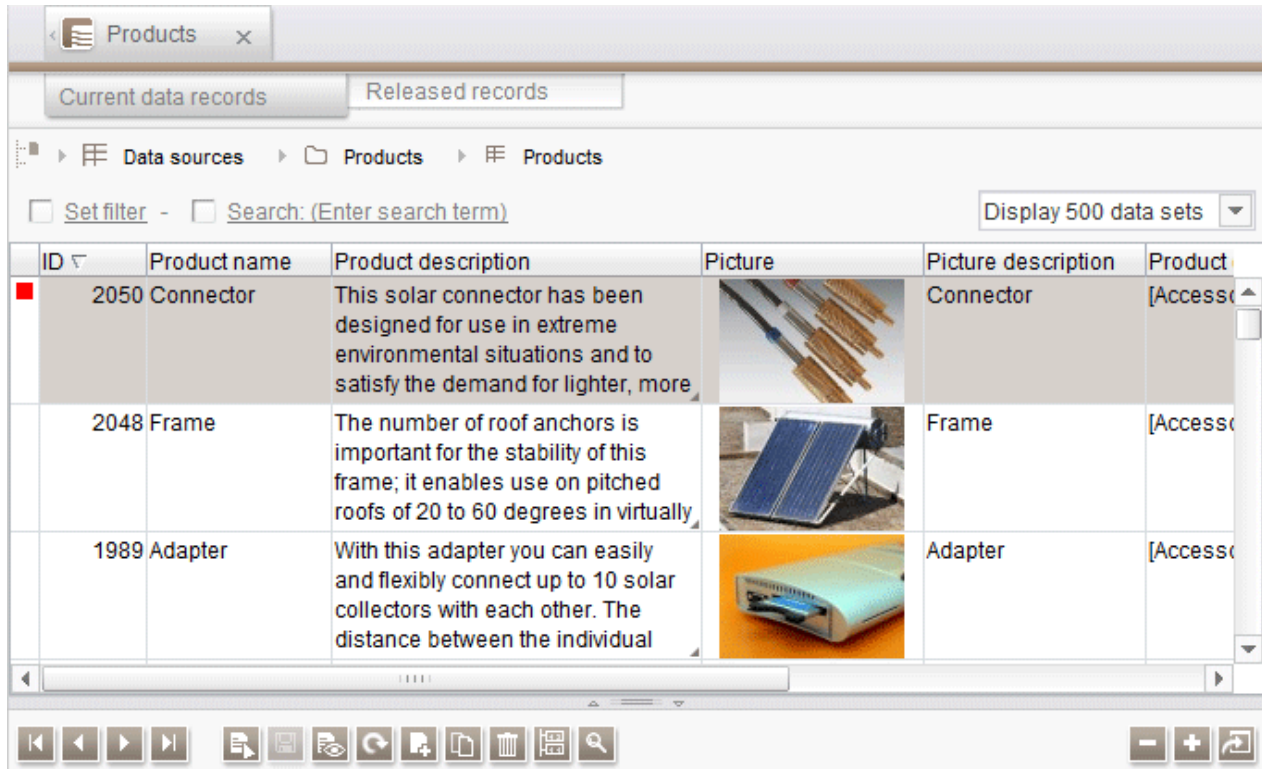


Figure 5-9: Table overview – Multi-line display

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 tool tip. Images referenced within the dataset are displayed as thumbnails.

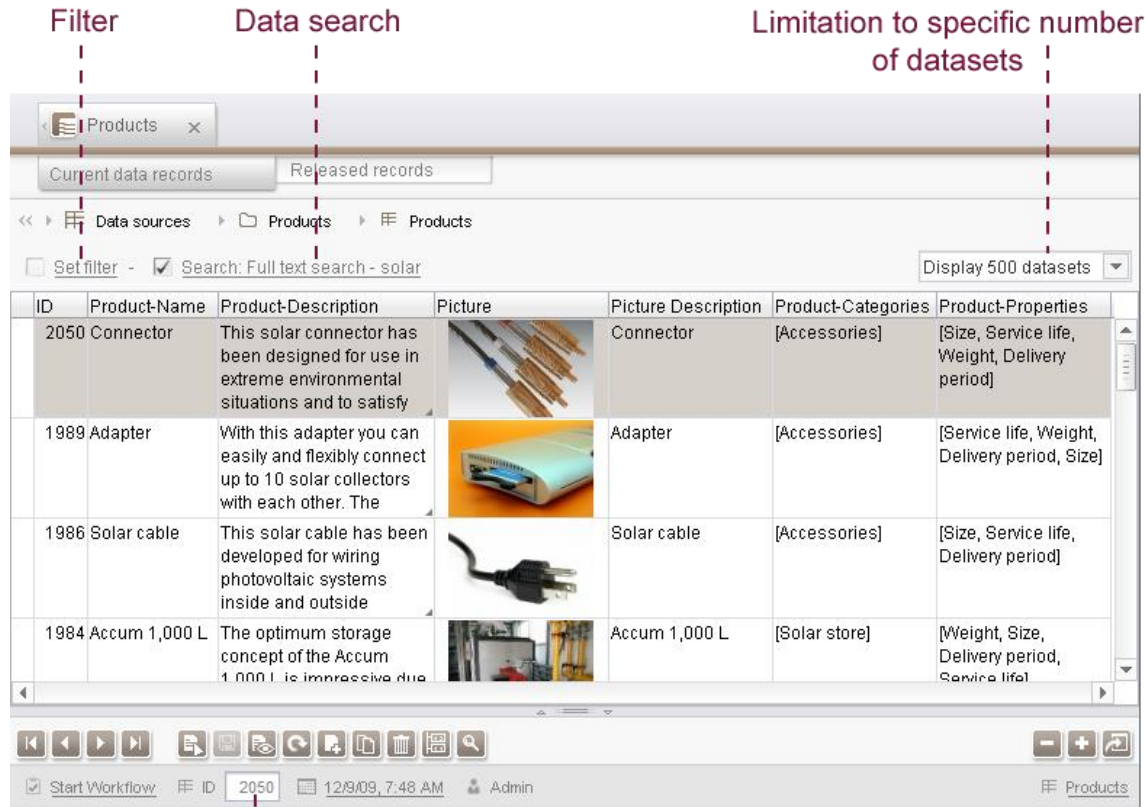
The line height, width and order of the columns can be pre-set by the template developer. This default view can be changed manually; however, the default view is restored when the view is updated.

5.5.2.2 Limit the number of datasets displayed

It is possible to set the number of datasets to be displayed in data sources. This makes work easier, especially when working with data sources with lots of datasets. This is done using a drop-down menu, located on the right above the datasets (see Figure 5-10). The number of datasets displayed can be restricted to 100, 500, 1000 or 2000. The number is restricted to 500




by default. In order to display all the datasets, select the entry "Display all datasets". The user merely needs "Visible" rights to use this function.



ID of the current dataset /  
Jump to dataset with specific ID

Figure 5-10: Data source with search, filter and limitation

The displayed datasets can be further limited by using filtering (see section 5.3.4 page 160) and the data search (see section 5.5.3 from page 168). The filter 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 the appropriate permissions.

All three options (filter, search, limit displayed datasets) can be combined with each other. If filtering or a search is active, this is indicated by a selected checkbox. These options are only **temporarily** valid, after <F5> or the corresponding icon in the tool bar is clicked (see section 3.2.4, page 78) 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 dataset is known, it can be entered in the input field in the status line. After pressing the <RETURN> key, the required dataset is displayed, provided any filtering that has been set allows this. As a default, the ID of the currently selected dataset is displayed in this field.

### 5.5.3 Dataset selection


The following buttons are available for navigation in existing datasets:

 First: click this icon to display the first dataset in the table.

 Previous: click this icon to display the previous dataset in the table.

 Next: click this icon to display the next dataset in the table.

 Last: click this icon to display the last dataset in the table.

 Data search: click this icon to open a search window in which specific datasets 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 sections. The search results are displayed in the main window. The "Update" icon in the tool bar or <F5> must be pressed to display all the available datasets.

#### 5.5.3.1 Full text search

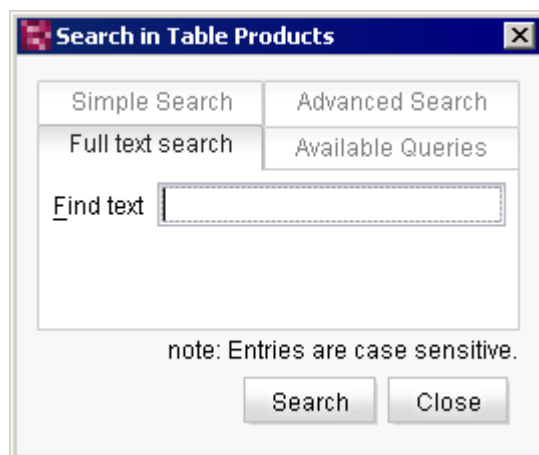


Figure 5-11: Full text search



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 Rich Text Editor. To do this, a search term is entered in the "Find 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. Foreign key relations with other tables are not taken into account.

### 5.5.3.2 Available Queries

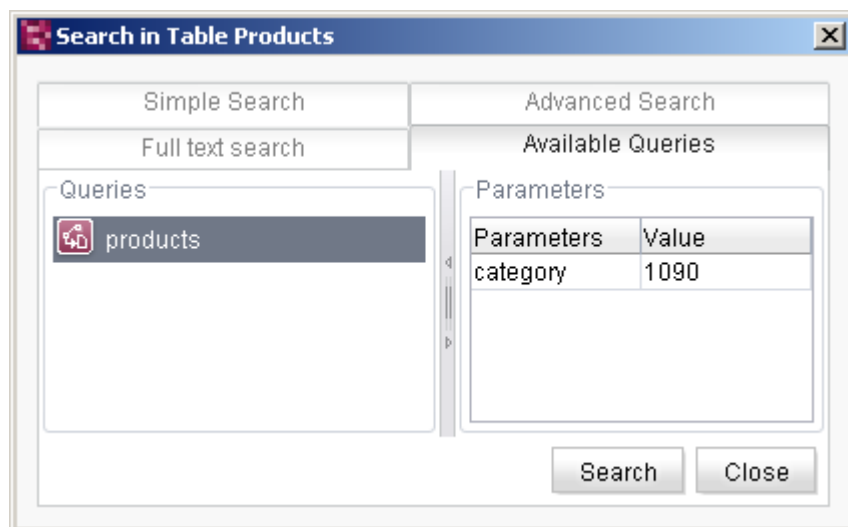
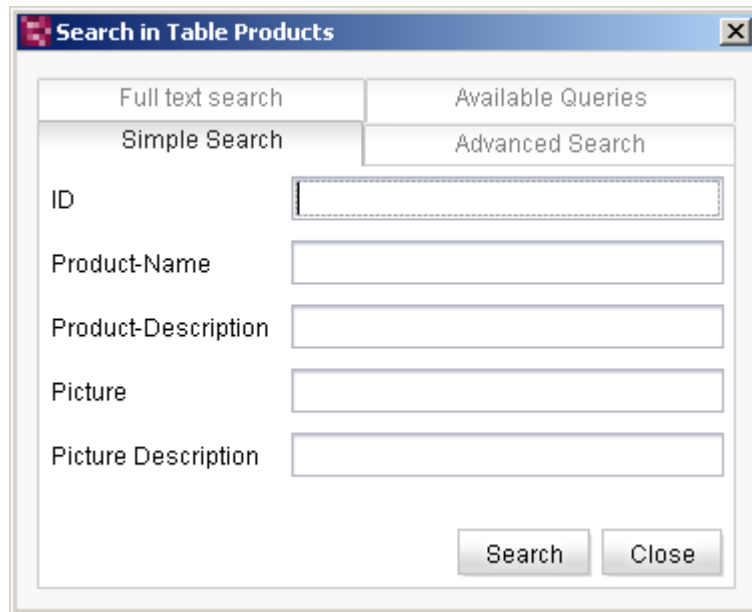


Figure 5-12: Available Queries

Prepared standard search queries can be used in the "Available Queries" tab. These queries have been created by the project developers to make subsequent searching for datasets easier. You simply have to select one of the available queries and activate the **Search** button. The search results appear in the main window.



## 5.5.3.3 Simple Search



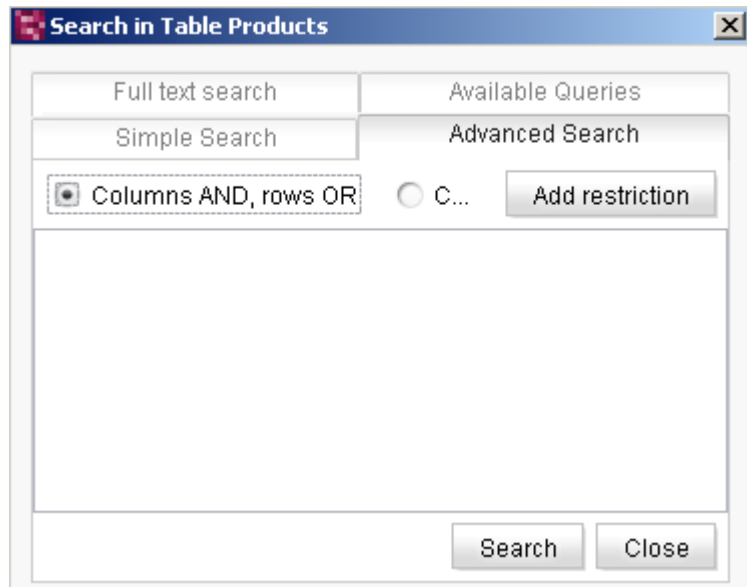
The screenshot shows a dialog box titled "Search in Table Products". It has a close button in the top right corner. The dialog is divided into two tabs: "Simple Search" (which is selected) and "Advanced Search". Under the "Simple Search" tab, there are five input fields with labels: "ID", "Product-Name", "Product-Description", "Picture", and "Picture Description". The "ID" field is currently empty. At the bottom right of the dialog, there are two buttons: "Search" and "Close".

**Figure 5-13: Simple Search**

The "Simple Search" tab can be used to search for specific content in one of the input fields of a dataset. 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-text search is performed for searches in other form fields of the datasets. References contained in the datasets are also searched through, e.g. references to media, page references, etc. Searches in input fields in which, for example, images are referenced, search through the respective reference names of the images. Terms can also be entered in several fields to further limit the search. The search results appear in the main window.



## 5.5.3.4 Advanced Search

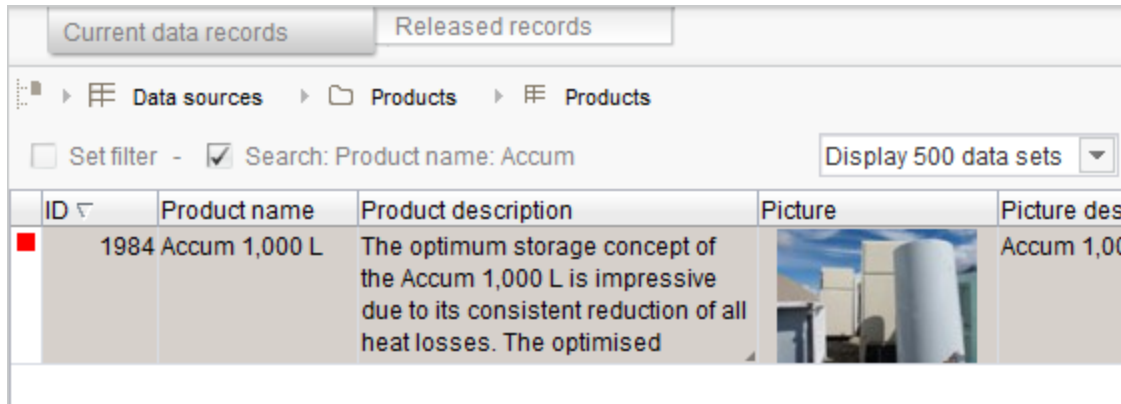
**Figure 5-14: Advanced Search**

The "Advanced Search" tab can be used on the one hand to create your own queries similar to the "Available Queries" (see section 5.5.3.2, page 169). 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 what are known as "foreign keys". *A precise description of the "Advanced Search" is given in the Query section of the section on database schemata in the Documentation for Developers manual.*



### 5.5.3.5 Presentation of the search results

The activated search is displayed above the list; in the example in Figure 5-15, this is a search for the product name "Accum".



The screenshot shows a web interface with two tabs: 'Current data records' and 'Released records'. Below the tabs is a breadcrumb trail: 'Data sources > Products > Products'. There is a search filter section with a checkbox 'Set filter' (unchecked) and a checked checkbox 'Search: Product name: Accum'. To the right of the search filter is a dropdown menu 'Display 500 data sets'. Below this is a table with the following data:







ID	Product name	Product description	Picture	Picture des
1984	Accum 1,000 L	The optimum storage concept of the Accum 1,000 L is impressive due to its consistent reduction of all heat losses. The optimised		Accum 1,00

Figure 5-15: Search results

**Checkbox search:** This checkbox can be used to quickly activated and deactivate the search so that only the datasets to which the search criteria apply or all datasets can be displayed.

### 5.5.4 Data entry

The bottom area for data entry is displayed by clicking the icons: , , ,  or . The fields of the datasets 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 tool bar up or down, or use the +/- icons. Double-click an existing dataset in the data overview to open the detailed view in a separate window. The same icons for dataset navigation and editing are available in the detailed view.

During editing, the currently edited dataset is displayed within the overview with stronger highlighting and in bold. In this way, the editor can recognize at a glance which dataset is currently being changed. If the editor selects a new dataset in the overview, the focus in the bottom editing area changes and the now current dataset is highlighted bold in the overview.



**Edit dataset:** click this icon to change an existing dataset. Similar to the other stores, Editing mode is switched on as a result and the icon is displayed "pressed in". A renewed click on the icon switches Editing mode back off again, any changes made in the meantime are saved. If Editing mode is switched on for a dataset, it remains switched on even if other datasets of the



same data source are selected. This makes it possible to edit several datasets in series.



*If a dataset which is in Editing mode and has been changed is quit, e.g. by selecting another dataset 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 dataset and the values of a new dataset. Editing mode is not quit. The icon is only active if changes have been made to the dataset.



*When editing and saving datasets, two different methods are possible, depending on the project administrator's settings.*

### **1. Parallel working**





*Unlike elsewhere, the data nodes are not blocked against data entry. Therefore, various datasets 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 dataset 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 dataset are accepted initially. If the second editor now tries to save the changes to the dataset, 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 dataset to display the changes of the second editor or save their own changes despite this and therefore overwrite the changes of the other editor.*

### **2. Exclusive working**

*If **exclusive editing of datasets** is activated, a dataset can be switched to Editing mode so that no other editors can make any changes to the dataset.*

*The ID and text of a dataset currently in Editing mode are shown in bold. Only one dataset can be in Editing mode at any one time. If you switch to another dataset (dataset 2) using the cursor or the icons    or , Editing mode is deactivated for the first dataset (dataset 1). If changes were made to dataset 1 and they have not been saved, the query "Do you want to save the changes?" appears. If you want to edit the new dataset (dataset 2), you have to switch it back to Editing mode. This means it is no longer possible to edit several datasets in series.*

*If a dataset is currently being edited by another editor, information to this effect appears and the dataset cannot be edited.*





Preview: click this icon to display a preview of the selected dataset.



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



*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 tool bar, unsaved changes are automatically saved first*



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



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



New (copy data): this icon is used to create a new dataset as a copy of the selected dataset. The values of the copied dataset are adopted for the new dataset. 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 you have selected more than one single dataset at once, only one confirmation prompt will be shown for all datasets. If you confirm, all selected datasets will be deleted.*





*If a dataset is to be completely deleted from the data source, it must also be deleted with released status (see section 5.5.5 page 175). All deleted datasets can be restored again at any time using the "Restore Deleted Objects" function in the context menu of the respective data source (see section 5.5.5 page 175).*



Version management: click this icon to open a window listing all versions of the current dataset. For precise documentation of the version history in the Content Store, see section 11.10.4, page 382.



Data search: click this icon to open a search window which can be used to search through the table of the current datasets. The different types of data search are explained in sections 5.5.3.1 to 5.5.3.4, from page 168.



Minimize/Maximize: click this icon to reduce or increase the size of the editing area.



Click this icon to open the selected dataset in a separate editing window.

### 5.5.5 Data overview – Released records

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





ID	Headline	Subheadline	Teaser	Date	
704	Achievable optimum	The importance of mains conne...	You should definitel...	21.11.2008	Z
256	Mithras Energy again awar...	The thin film modules have now ...	As in 2007, Mithras ...	02.01.2009	D
132	New product range	Thin film modules are gaining g...	The thin film modul...	13.10.2008	Z
131	Solar diversity	Widespread use of mono-crysta...	Mono-crystalline m...	27.05.2008	W
130	New Director of Mithras En...	Hans Energie reinforces the exe...	Since 1 January 20...	15.01.2008	H
128	Mithras Energy receives s...	Constant innovation pays	Mithras Energy was...	30.12.2007	M

**Figure 5-16: Table overview – Released datasets**

The display of the table overview and the navigation or dataset selection corresponds to that of the current datasets (see section 5.5.2 page 164).



*You can use the first column to sort released datasets by the time of their last release. This way, you can click to display datasets that were released recently at the top of the list.*



**Update:** icon can be used to update the data overview of the released datasets if necessary, because new datasets may have been released in the meantime.



**Delete:** datasets which only exist with released status are identified by gray marking in the first column. Click this icon to delete the selected dataset with released status. All deleted datasets can be restored again at any time using the "Restore Deleted Objects" function in the context menu of the respective data source.








**Data search:** click this icon to open a search window which can be used to search through the table of released datasets. The different types of data search are explained in sections 5.5.3.1 to 5.5.3.4, from page 168.



## 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 images
-  Language-dependent images
-  Language-independent files
-  Language-dependent files

Specific icons based on the file type are displayed for files. In this way, the data type is immediately apparent in the tree structure, without having to change to the detailed view first.

### 6.1 General Media Store context menus

The Media Store context menus are described in the following sections:

For general information on handling context menus see section 4.1 General context menus in the Page Store page 128.

#### 6.1.1 New

The "New" context menu entry can be used to insert new objects into the project.

**New – Create 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 6-1: New – Create 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.

A **language-dependent display name** can be assigned to the new medium, 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 "View – Preferred Display Language" menu (see section 3.1.5.2 page 55). 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.



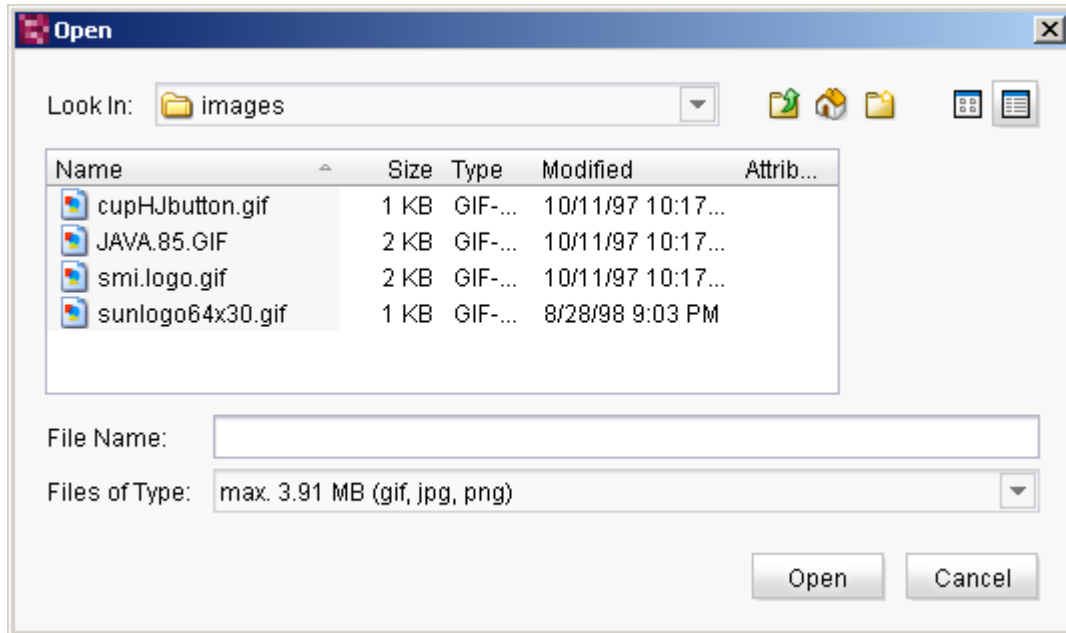
*The Reference Name field is only displayed if this setting was configured accordingly by the project administrator.*

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.

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*).



**Figure 6-2: File selection dialog with media restriction**

Media that are larger than the maximum size defined in the project properties and/or have a file name extension which is not allowed, are 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 6-2, files with file name extensions "jpg", "png" and "gif" up to a size of 3.91 MB may be selected and uploaded. Click "Open" to copy the selected medium into the dialog in Figure 6-1.



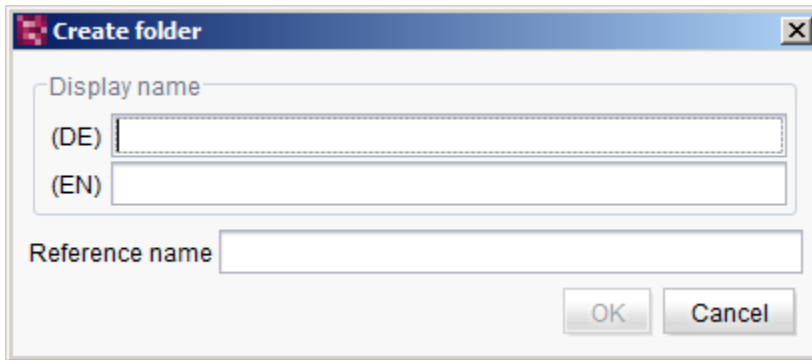
*A corresponding warning appears if files from the workstation computer's directory structure that are not allowed due on the basis of the media restrictions are added to the Media Store by means of "drag-and-drop". The files are not created in the Media Store.*

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 6-3: 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 "View – Preferred Display Language" menu (see section 3.1.5.2 page 55). 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.

If this function is called on a medium, the new folder is always inserted beneath the folder containing the medium.



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



### 6.1.2 Editing on/off

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

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

### 6.1.3 Reset Changes

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

### 6.1.4 Cut

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

### 6.1.5 Copy

This function is used to generate a copy of the current object and store it in the clipboard. Copies can be generated from images and files.

### 6.1.6 Paste

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

### 6.1.7 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.



### 6.1.8 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, for example for deleting objects, can be tied to this function (see section 3.2.9.7, page 87). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

### 6.1.9 Display in current/new workspace

These functions allow you to choose the number of open tabs in the editing area of the FirstSpirit Client.

**Display in current workspace** opens the selected object in the central part of the screen on the active tab. **Display in new workspace** opens the selected object in a new tab.

### 6.1.10 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 section 11.10.5 page 383.

### 6.1.11 Workflow

If no workflow is 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 or state under this menu item.

Detailed documentation of workflows is given in section 12 page 390.



### 6.1.12 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.

## 6.2 Special Media Store context menus in the Media Store

### 6.2.1 Add to Content transport feature

This license-dependent function allows you to conveniently reuse project content between projects.

A detailed description of this is given in the *FirstSpirit CorporateContent* module documentation.

### 6.2.2 Export (at folder and media level)

This function is available to administrators only.

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.



*The "Export" function is a client-side function and therefore puts great strain on 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.*

### 6.2.3 Import (at root and folder level)

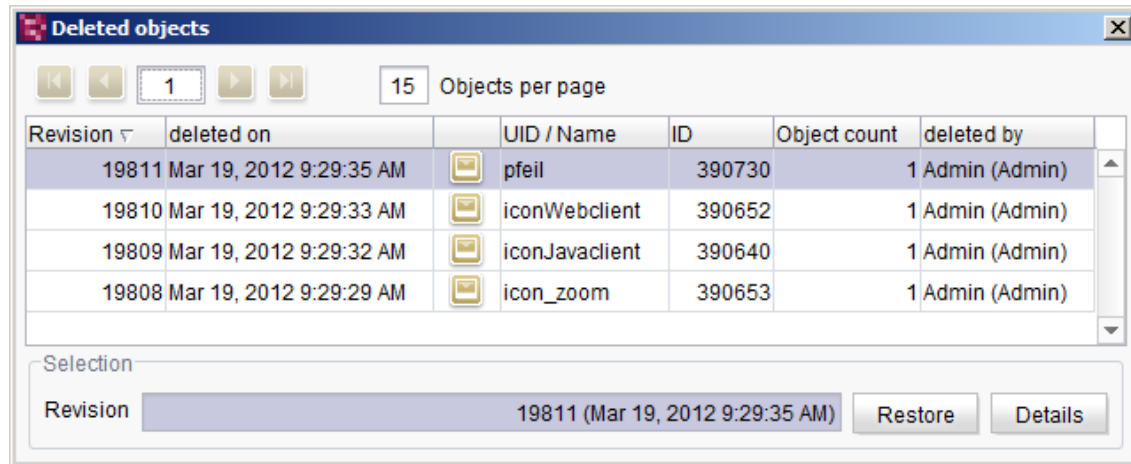
This function can be used to import an export file with all the necessary information back into the project. A window opens with the file structure of the computer, enabling you to select the required export file.





## 6.2.4 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 you click it, a window opens with the deleted objects.



**Figure 6-4: 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.

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

**Type of object:** Object icon indicating the type of object that was deleted.

**Name:** The reference name of the deleted object.

**ID:** The unique ID number 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 by:** Name of the user who deleted the object.

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



Restoration options and the position of the object to be restored can be determined in two further dialogs.

## 6.3 Functions under the Extras context menu of the Media Store

### 6.3.1 Release

This function is only available for administrators.

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 section 12.4 page 399.

### 6.3.2 Delete metadata

This function is only available for administrators.

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

### 6.3.3 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 section 13.1.2, page 409.

### 6.3.4 Reset write protection

This function is available to administrators only.

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

### 6.3.5 Show usages (at media level)

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.



### 6.3.6 Display properties

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. This function can also be called using the keyboard shortcut "ALT + P". The information can vary, depending on the object type.

*For further information about this function see section 4.3.11, page 142.*

### 6.3.7 Make medium language-dependent/language-independent (at media level)

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!

### 6.3.8 Cancel editing

Using this function the Editing mode can be ended without accepting changes which have been not saved yet. The status of media that have already been released is unaffected by this action.

### 6.3.9 Change reference name

This function is only available for administrators.

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

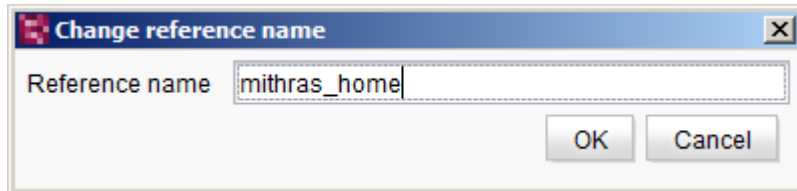


*Reference names are only shown in the project if the "Display reference names in tree" checkbox and the menu item View – Preferred display language have been activated. (See section 3.1.5.2, page 55)*

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 name" 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. an image 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 6-5: Change reference name**

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

Depending on the project administrator's settings, the menu entry may be grayed out regardless of the "Change" permission; in this case the reference name cannot be changed.

### 6.3.10 Reset stored URLs

This function is available to project administrators only.

If a path generation method which stores the generated URLs was used for generating the project (PathGeneration with "(SEO)" in generation schedules, see also *FirstSpirit Documentation for Administrators*, Chapter "Generation"), these URLs remain valid by default, even if the values of relevance for the URL generation process (depending on the path generation method selected, for example changed display or reference names, relocated pages or media in a tree, changes in datasets) should change. You can reset stored URLs by using this function so that changes in the tree structure or in display names will be reflected in the URL. See also Chapter 9.3 page 260.

This function affects elements located in the tree structure below the element on which the function was called. For example, you can reset all the stored URLs in a subtree by calling this function at the top menu level from which you want to reset the URLs. The stored URLs for all the page references and menu levels beneath this menu level are then reset.



### 6.3.11 Display dependencies (at media level)

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

For a detailed documentation of the reference graph please refer to section 11.11 page 386.

## 6.4 Editing area at folder level

The same view and setting options are available in the editing for the root of the Media Store, as in the folders of the Media Store.

### 6.4.1 Overview tab

All folders and media located in this folder are displayed in the Overview tab.

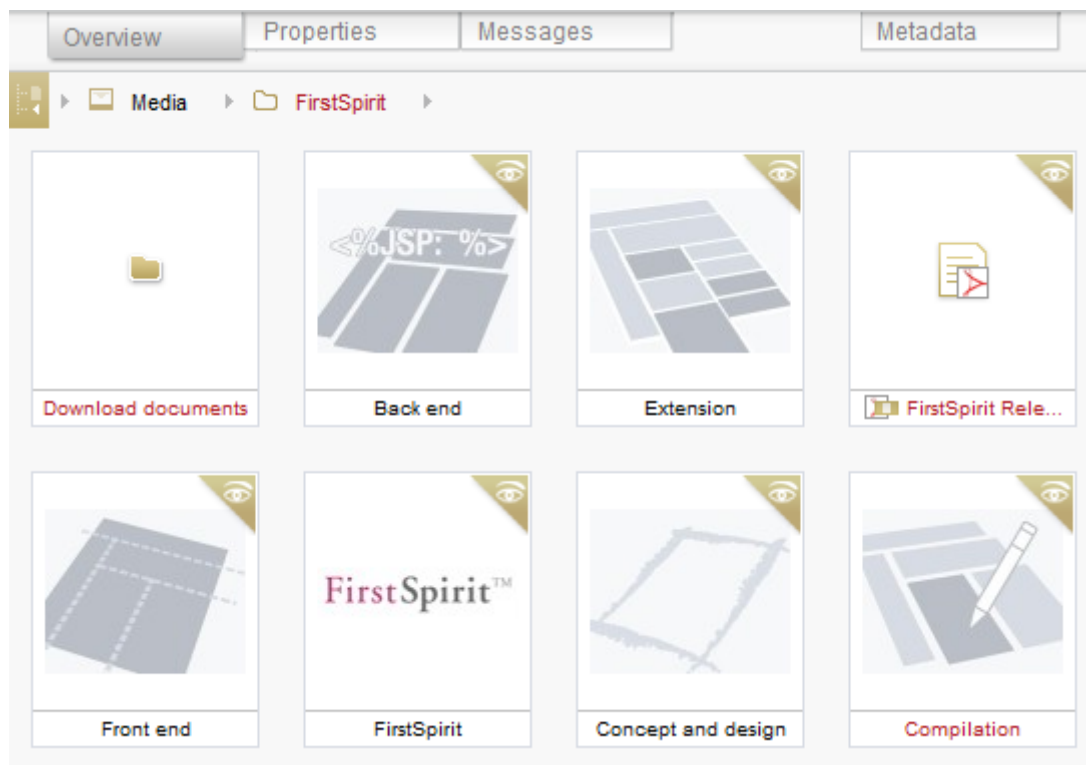





Figure 6-6: Folders view – Overview tab

Specific icons based on the file type are displayed for files.



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 section 3.1.5.5 page 59), the  icon indicates that the file concerned can be displayed in the Integrated preview (see section 3.4.2 page 116). The medium will open in the integrated preview by one click on this icon.

## 6.4.2 Properties tab

Various settings can be specified in the Properties tab.

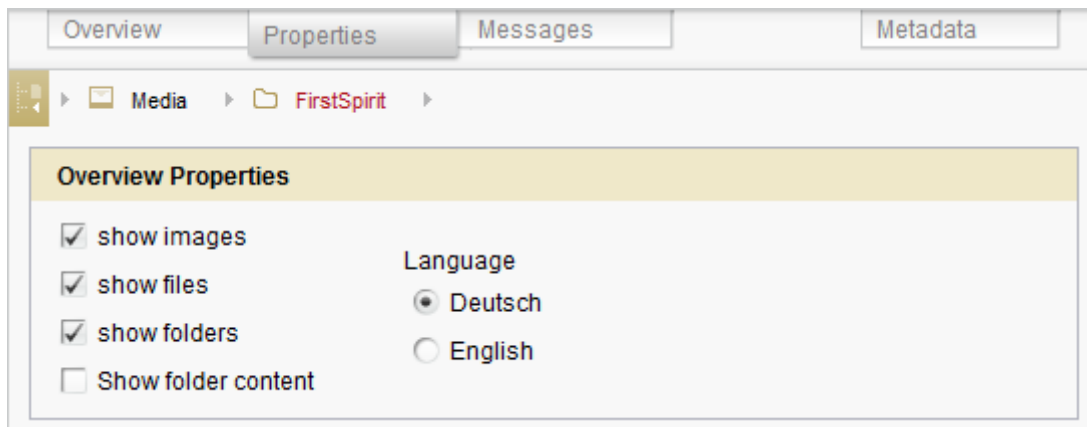


Figure 6-7: Folders view – Properties tab

Here you can set which content is to be displayed in the Overview tab.

**Show images:** All images in this folder are displayed.

**Show files:** All files in this folder are displayed.

**Show 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 images 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.*

### 6.4.3 Messages tab

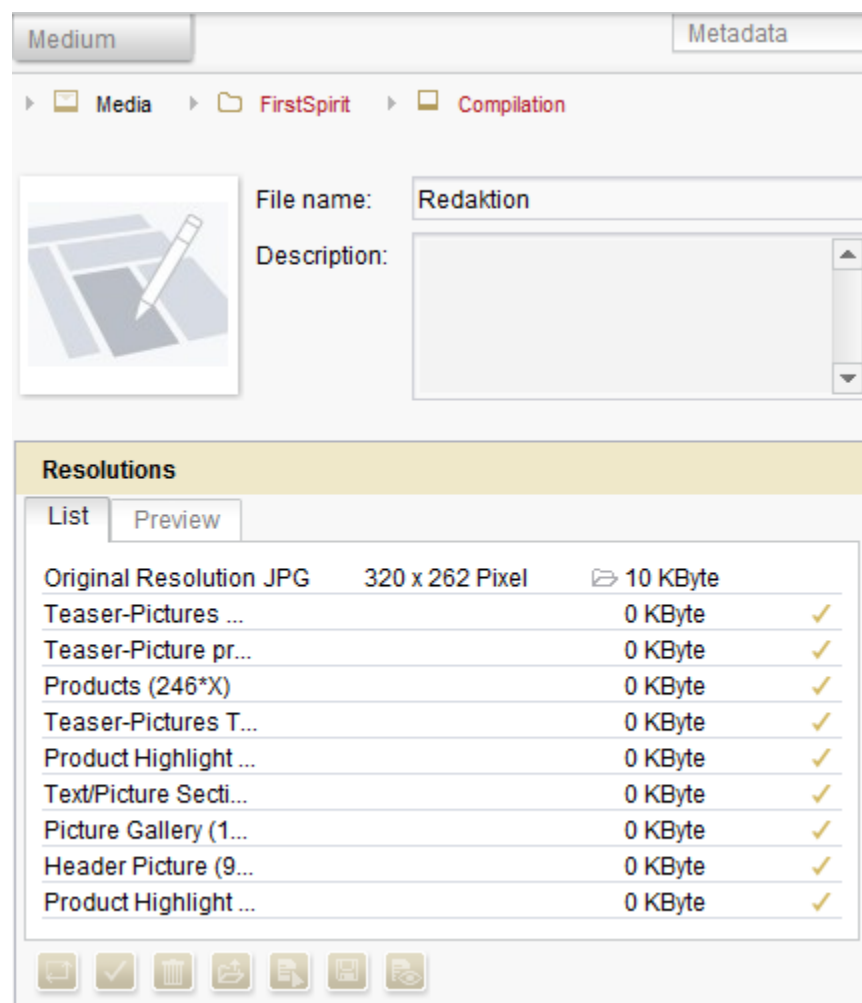
The Messages tab contains a message board.

Detailed documentation on use of message boards is given in section 11.2 "The message board" page 307.

## 6.5 Editing area for images

The setting options for language-dependent and language-independent images are identical. For language-dependent images there is a separate tab for each project language in which the same setting options are available again.





**Figure 6-8: Medium view – Image**

A preview (thumbnail) of the current image is automatically displayed in the top left-hand corner. If the Preview icon is clicked the program first tries to start the image 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 image on the website.

**Description:** This field can be used to enter an explanatory comment on this image which, among other things, can be used as a tool tip on the website. The explanatory comment on the image should of course be written in the relevant language.

The **Resolutions** area contains the options for changing the current image. The top row describes the original images, as can be seen in the thumbnail. In addition, there is information





on the image type, the image size and the file size.

At the very bottom, there is a row of icons that can be used to carry out certain editing functions for the individual resolutions. These functions are described in detail in section 6.6.1, page 194.

### 6.5.1 List tab

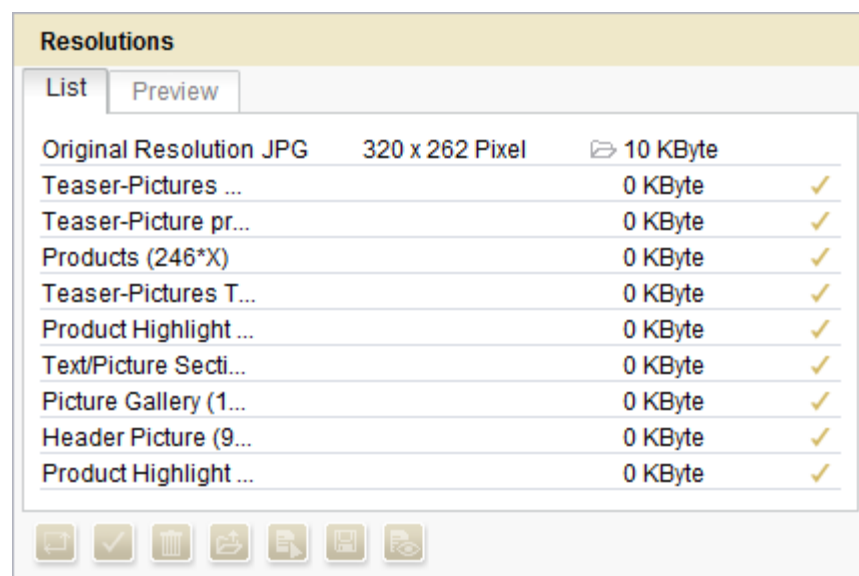


Figure 6-9: Medium – Image, List tab view

The resolutions defined in the project properties (see "FirstSpirit Manual for Administrators") are listed on the "List" tab.

The first line describes the original image, as can be seen in the thumbnail. In addition, the image type (here: JPG file), the image size (here: 320 x 262 points) and the file size (here: 10 kB) are displayed.

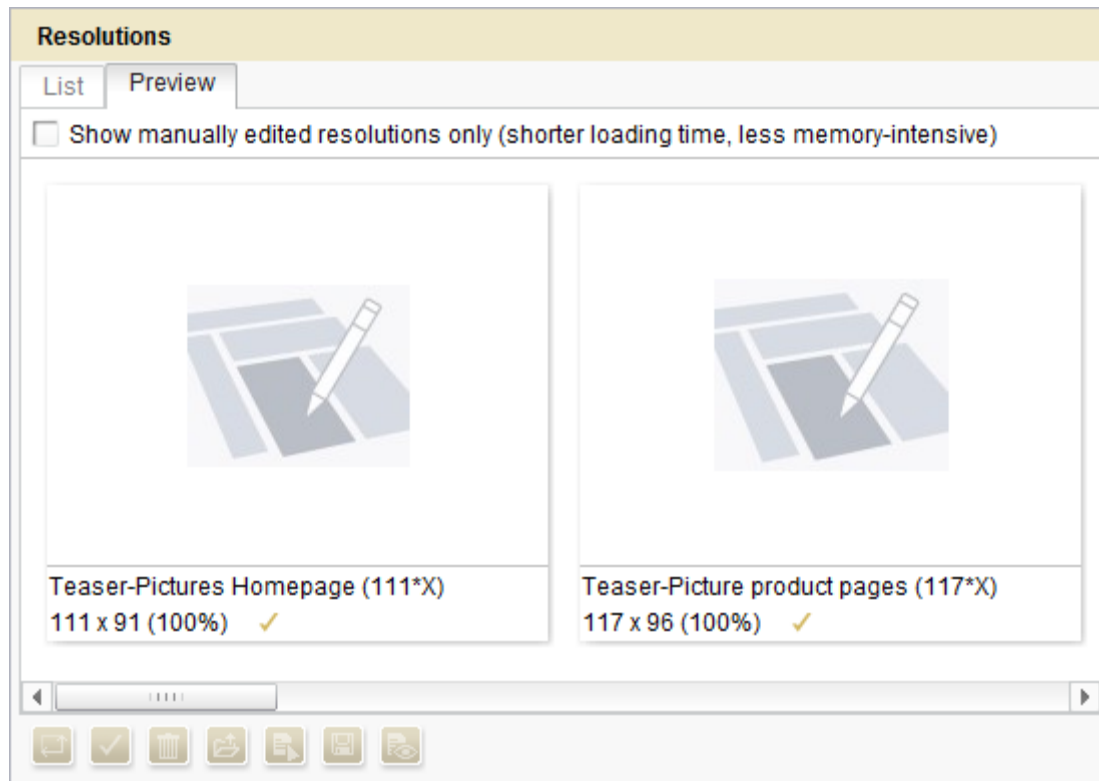
The other resolutions are calculated automatically by the system when needed. It is also possible to upload user images at any resolution or to determine image sections (see section 6.6.1, page 194). Each edited or uploaded image is also displayed with the image type, image size and file size, as well as the relevant status (✓, ⚠ or ?).

### 6.5.2 Preview tab

The Preview tab is only displayed if the integrated preview is **not** used for media. If the integrated preview for media is active, the information from this tab is displayed in the integrated preview area. (For more about the integrated preview, see section 3.1.5.5, page 59 and section 3.4.2,



page 116).



**Figure 6-10: Medium – Image, Preview tab view**

**Show manually edited resolutions only:** This checkbox is activated by default. A preview of each manually edited resolution is displayed, for example to enable the display to be more easily checked.

If this checkbox is deactivated, a preview is displayed for all resolutions (edited **and** automatically generated).

The identifier of the relevant resolution, the dimensions and the relevant status (✓, ! or ?) are specified for each preview image. A percentage smaller than 100 indicates that the preview image is displayed with reduced size. Double-click the preview image to display it in a pop-up window with 1:1 display.

## 6.6 Edit images

FirstSpirit provides various options for making simple changes to images from the Media Store without having to open them in external image editing software. In this way, images can be, e.g. resized, rotated or mirrored.



### 6.6.1 Functions in the editing area

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.





Crop image: click this icon to open an editing window in which you can edit the resolution (see section 6.6.2, page 197).



*The original image 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 image of a cropped image has been subsequently replaced. The resolution of the cropped image is not automatically adjusted to the size of the new original image 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 image or editing in the external editor); the resolution is rendered once again from the original image. The original image itself cannot be deleted with this icon.



Select new image: click this icon to open a window in which you can move through the file structure at your workstation to search for the required image.



*If the size of the uploaded image does not match the size of the specified resolution, the  icon is displayed in the list. The image is then output on the website exactly the same as it was uploaded.*



Edit, click this icon to open the current image 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 image is opened. After the changes have been made in the external editor, the image 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 elements opened externally are listed.



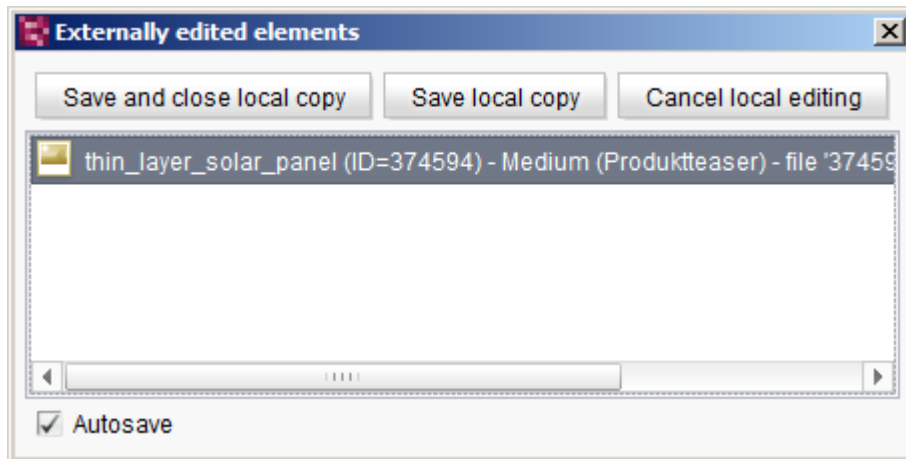



Figure 6-11: Externally edited elements

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 JavaClient. (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 JavaClient 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 JavaClient.



*If the size of the edited image does not match the size of the specified resolution, the  icon is displayed in the list. The image 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 FirstSpirit Client, click this icon to copy the edited image 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:**















 Cropped: this icon indicates that the image has been edited using the dialog in Figure 6-12.

 Uploaded: this icon indicates that the image has been uploaded using the  icon or edited using the  icon.

 Checked: this icon indicates that the deposited image corresponds to the predefined resolution or can be automatically rendered by the system.

 Wrong resolution: this icon indicates that in the case of images uploaded using the  icon or edited using the  icon, the deposited image does not correspond to the predefined resolution. The image 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 image was replaced after the resizing. The resized resolution should be checked and if necessary adjusted. Several options are available for this:


- : the image is edited internally: In this case the original image is used. After saving, the  icon in the list of resolutions becomes a .
- : a new image is uploaded. If the dimensions of the uploaded image match the resolution, the  icon in the list of resolutions becomes a . Otherwise a  is displayed.
- : the image is edited in an external editor. The resolution of the edited image 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 that were not edited beforehand are automatically re-rendered with the new original image and displayed if the original image has been replaced.*

## 6.6.2 Integrated cropping function

If the cropping function is called using the  icon, the system first checks whether a resolution already exists and whether it is based on the original image or another image:

- If no resolution exists yet, the original image is opened for editing in the following dialog (see Figure 6-12).
- If a resolution exists which is based on the original image, the original image is also opened for editing in the following dialog (see Figure 6-12).
- If a resolution exists which is not based on the original image but on an uploaded image, a message appears: "This resolution is not based on the original medium. Further resizing can cause loss in quality.", and then the uploaded image is opened for editing in the following dialog (see Figure 6-12).



*In this case, only the resized image is saved, the sizing can only be made smaller at a later date.*

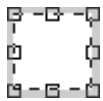




**Figure 6-12: Resizing function for images**

The selected image can now be edited in this dialog. Apart from the image name, the current image size is also displayed (here: 848 x 565 px).

The following editing options are available:



If the image is larger than the target resolution, the frame can be used to select a suitable cut-out from the image. The first time the image is edited, as a default the frame is displayed centered with the size of the target resolution. The image area outside the frame has a gray background. If the image 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 image. 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 (see *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 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 image by simultaneously pressing the Shift key.



*If the original image is to be edited, a frame must be manually pulled onto it first.*

**Selection:** These pixel dimensions specify 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 image larger. If a larger cut-out is selected than the target resolution, the system automatically scales down the image to the resolution size.

**Target resolution:** These pixel dimensions specify the size defined for this resolution in the project properties (see *FirstSpirit Manual for Administrators*).



Rotate 90° to the left: this icon can be used to rotate the image through 90° to the left. If necessary, the selection frame must be moved or reduced in size within the image so that the selection can be accepted.



Rotate 90° to the right: this icon can be used to rotate the image through 90° to the right. If necessary, the selection frame must be moved or reduced in size within the image so that the selection can be accepted.



Flip horizontally: this icon can be used to mirror the image horizontally.



Flip vertically: this icon can be used to mirror the image 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 image is smaller than the target resolution, this icon is inactive, as well as in the original image.*



Reduce zoom size: as a default, the size of the image 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 image to be viewed.




Increase zoom: this icon can be used to increase the zoom to examine a cut-out of the image 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 image.



, this combo box can be used to select predefined zoom levels. Use the "Fit to screen" to return to the initial zoom view.

The **Accept** button is used to save the image with its current edit status and the pop-up window closes. If the edited image is based on the original image, another image cut-out can be selected later, if necessary. If the image is not based on the original image, the resized image is saved. The sizing can only be made smaller at a later date. The same applies if the original image has been edited.

The image is now displayed in the list of resolutions with its file type, image size and file size. The  icon indicates that the image has been edited using the cropping function.



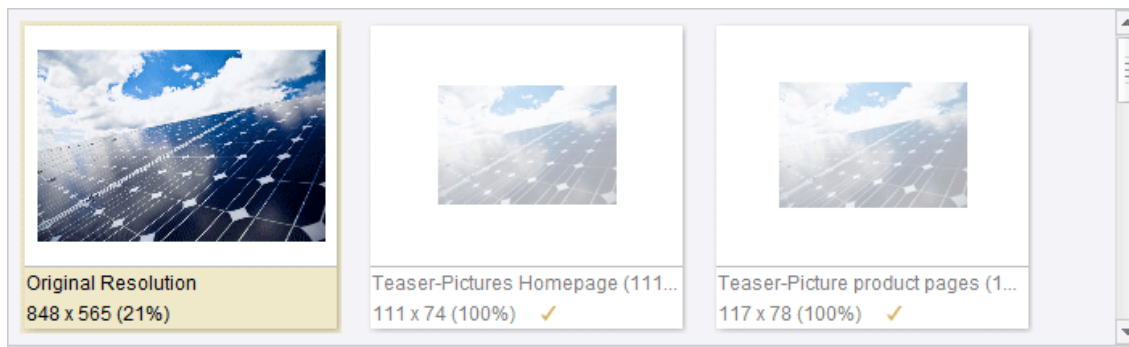
### 6.6.3 Processing of media in the Integrated preview



*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, section 3 "The FirstSpirit AppCenter".)*


In order to edit images in the Integrated preview, the image to be edited must be set in Editing mode 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 colors, which are automatically rendered by the system; resolutions which have not been edited are displayed in pale colors.





**Figure 6-13: 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 image 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 section 6.6.2 page 197).



Changes to images/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.

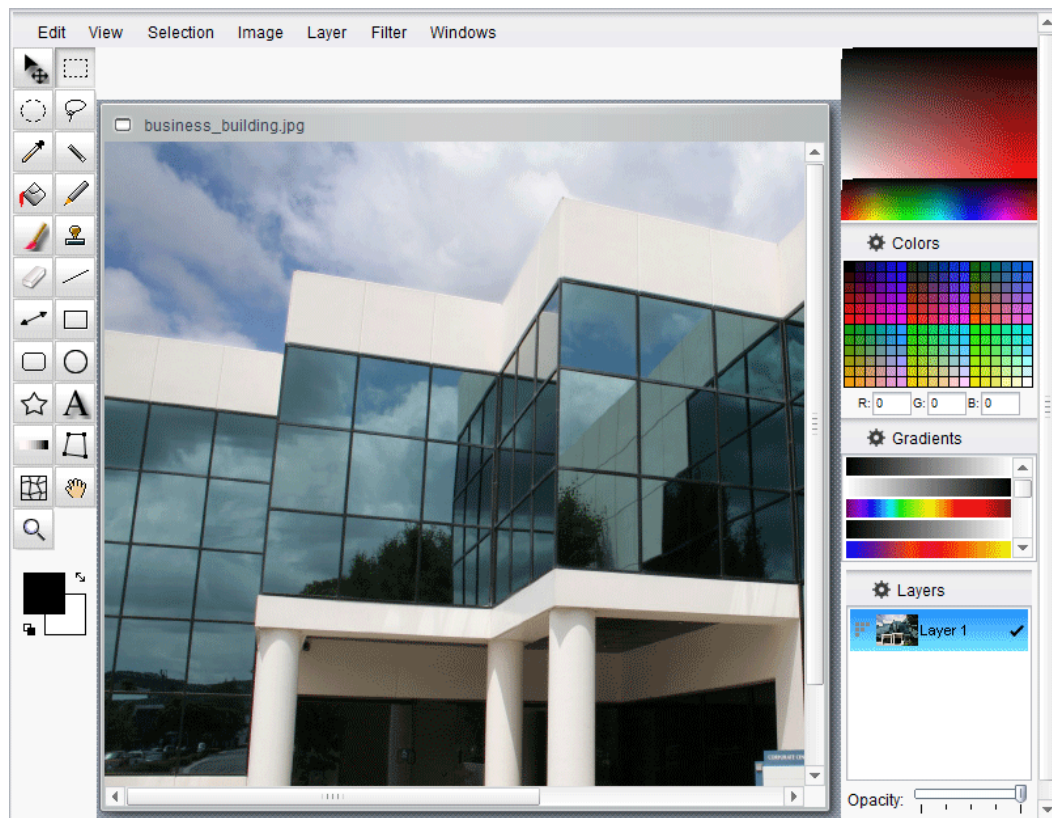
#### 6.6.4 Java Image Editor

If the "Graphic engine – Java Image Editor" option is activated, images 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. In other words, 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 6-14: Enhanced image editing – Java Image Editor**

Changes to images/resolutions first have to be copied into the Media Store of FirstSpirit by clicking the  icon. You can then continue editing the image. Finally, the changes must be saved using the "Save" or "Exit Editing" button.

Click the **Enhanced 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, tool tips and configuration dialogs (e.g. for filters) cannot be localized for technical reasons and are therefore only available in English.*





"Edit" menu, modified using 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 color.



Use this icon to select through the color of pixels in the image.

### Painting and drawing tools



Use this icon to fill adjacent areas with similar color.



This icon can be used to draw a hard-edged freehand line.



This icon is used to draw softer colored 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 the 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



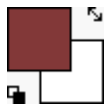
Use this icon to insert a layer with text into the image.



This icon is used to add color 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 color. They can be swapped using the arrow icon. Use the black&white icon to select black and white as the background/foreground color.

### 6.6.4.2 Menu levels

#### "Edit" menu

This menu is used to apply selections: They can be

- cut (**C**ut)
  - copied (**C**opy)
  - inserted (**P**aste)
  - deleted (**C**lear)
- 
- 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 (**U**ndo).

#### "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 color.

## "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.

#### 6.6.4.3 Palettes

- **Colors:** The foreground color can be selected here with a click.
- **RGB colors:** The foreground color can be selected here by setting RGB values.
- **Swatches ("Colors"):** Here the foreground color can be selected from a library. As a default, a palette of colors is used, which can be used, platform-independent, by all browsers ("web colors"). If necessary, you can import your own libraries (and save them again later). Uploaded color libraries are not saved on exiting the Editor and are no longer available the next time the Java Image Editor is started.
- **Gradients:** A color gradient can be selected here. The existing ones can be used, edited or the user's own gradients can be loaded. Uploaded or edited color 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 section 6.6.4.2 page 207. 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.



### 6.6.5 Simple image processing (PicMonkey)

If the "Graphic engine – Simple image processing (PicMonkey)" 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 images 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.

Changes to images/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 image. 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.

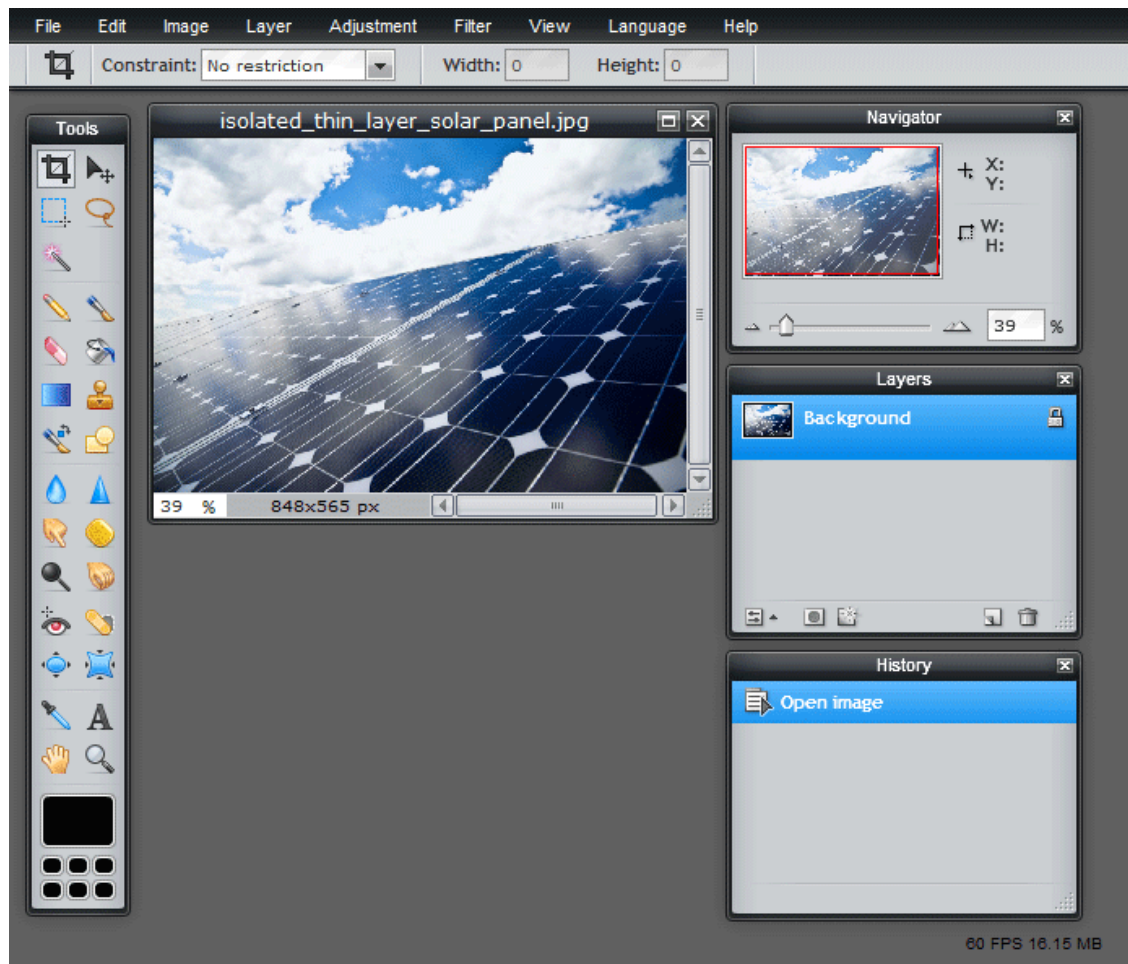
### 6.6.6 Enhanced image processing (Pixlr)

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 images 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 6-15: Enhanced image editing (Pixlr)**

Changes to images/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 image. Finally, the changes must be saved using the "Save" or "Exit Editing" button.

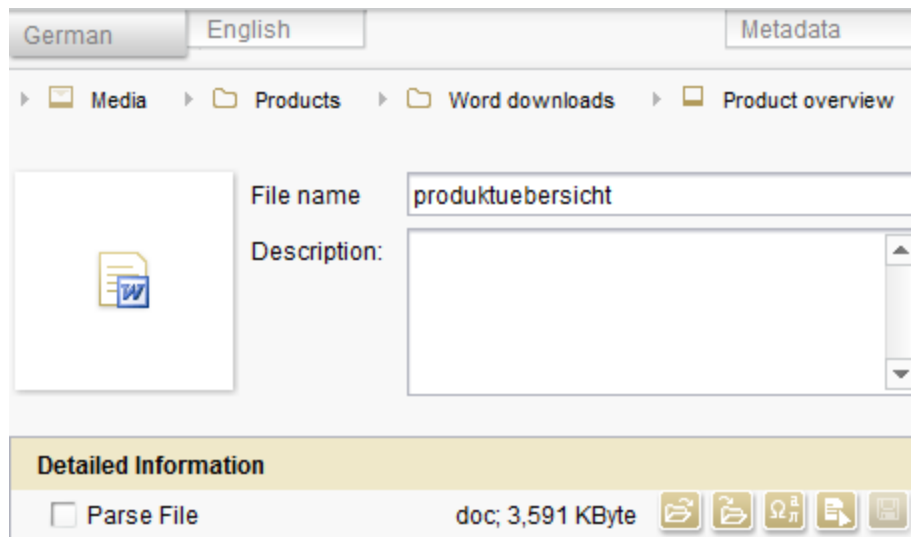
Click the Enhanced image editing button again to close the editor; any changes not saved are not adopted in FirstSpirit.



## 6.7 Editing area for files

Files means all other media formats apart from images. These include 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 6-16: 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.

The file only opens in an editor if the integrated preview is **not** used for media. If the integrated preview for media is active, the content of the file is displayed in the integrated preview area. (For more about the integrated preview, see section 3.1.5.5, page 59 and section 3.4.2, page 116).

**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: doc) and the file size (here: 3,592 kB). 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 country-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.



### 6.7.1 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 Editing 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.

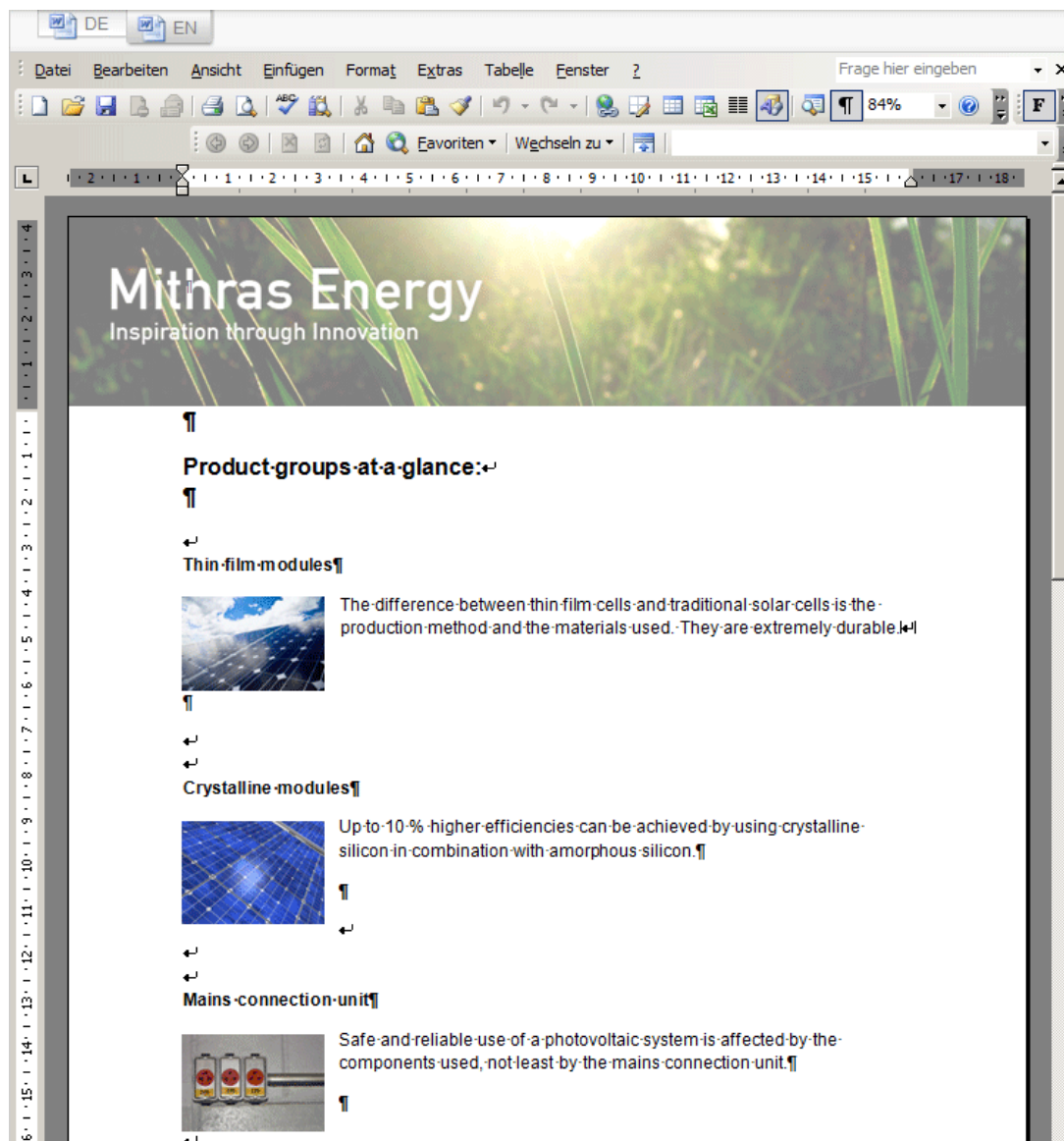


Figure 6-17: File editing with Microsoft Office



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.

## 6.7.2 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 Editing 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.

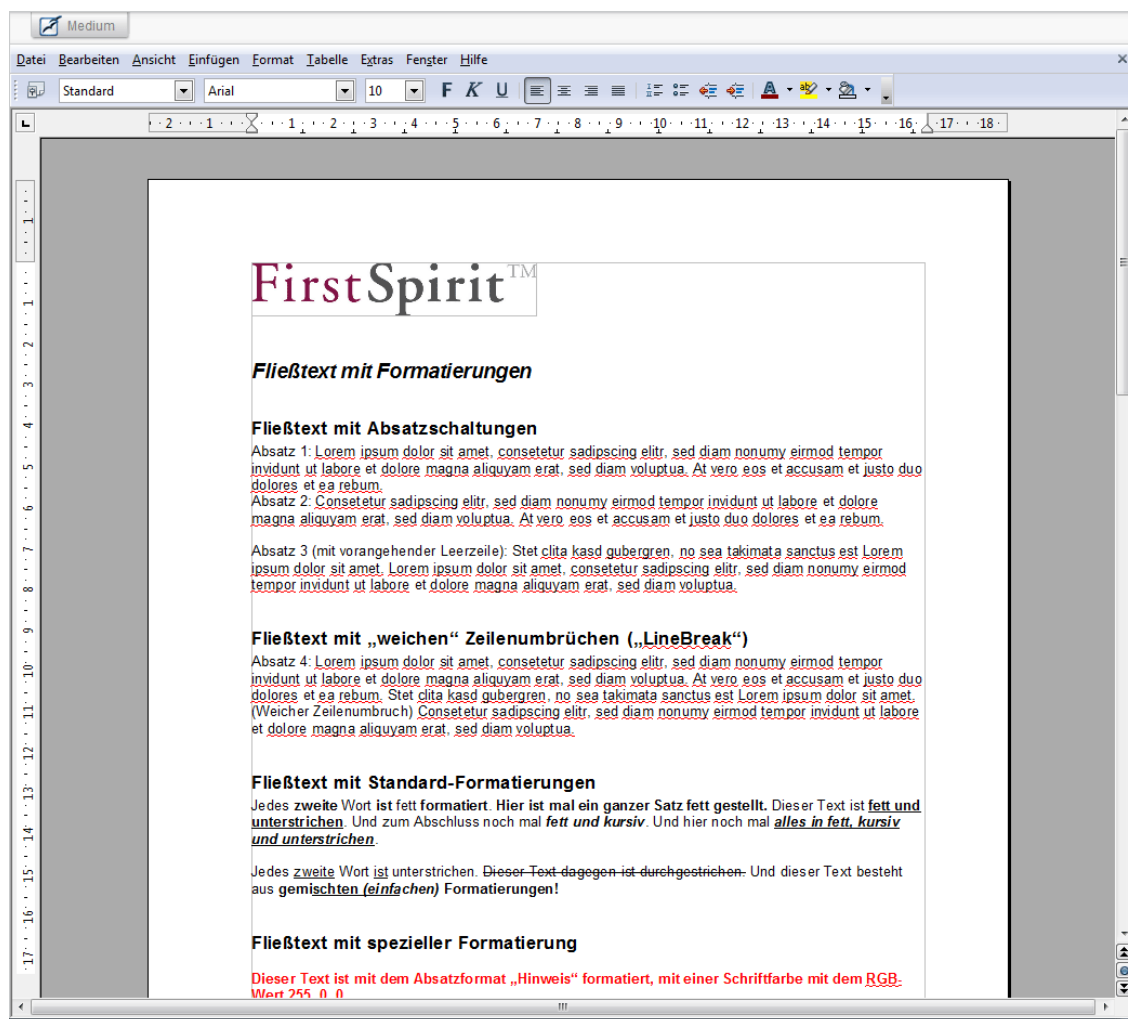


Figure 6-18: File editing with OpenOffice

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, for example if they 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

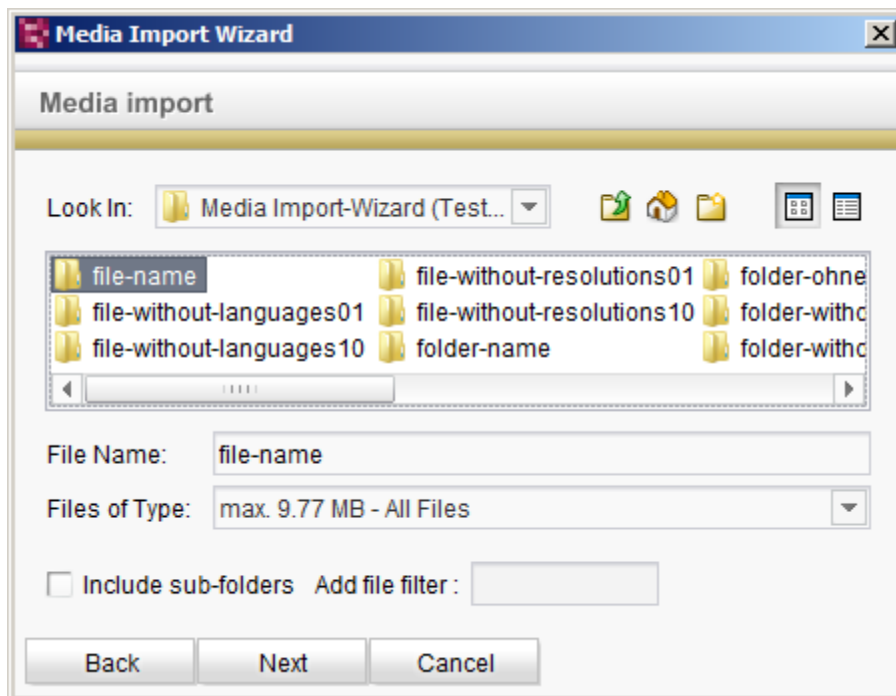


Figure 6-19: Media Import Wizard – Step 1

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!*





**Files of Type:** Here you can choose between All File Types and images (.jpg, .gif, .png)

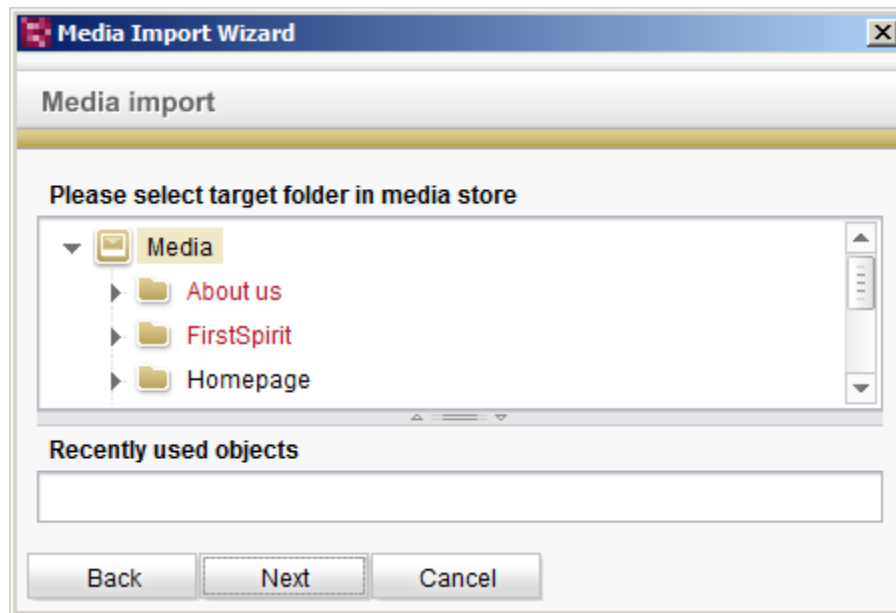


*Selection of media can be restricted to files of specific size and format according to the settings in the project configuration (see FirstSpirit Manual for Administrators). In this case media which are bigger than the maximum size which is defined in the project configuration and/or which have a file extension which is not allowed, are not imported by the Media Import Wizard.*

**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 6-20: 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

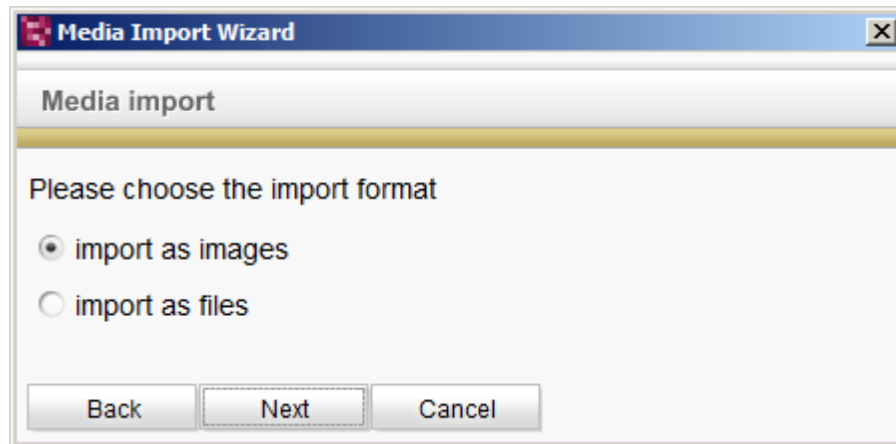


Figure 6-21: Media Import Wizard – Step 3

In the third step the user selects whether the imported media are to be created in the Media Store as images or as files.

### 6.8.4 Media Import Wizard – Step 4 of 9

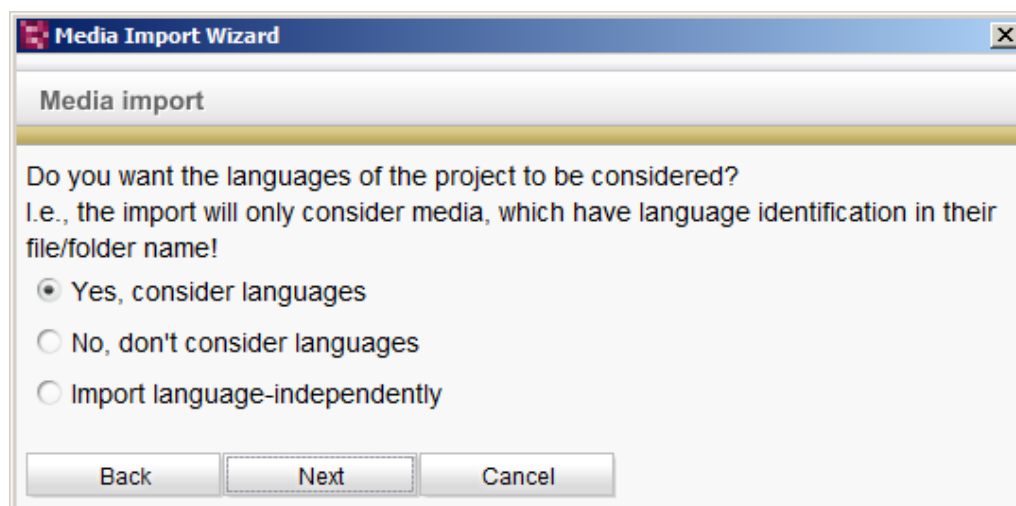


Figure 6-22: Media Import Wizard – Step 4



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, don't 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-independently:** 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

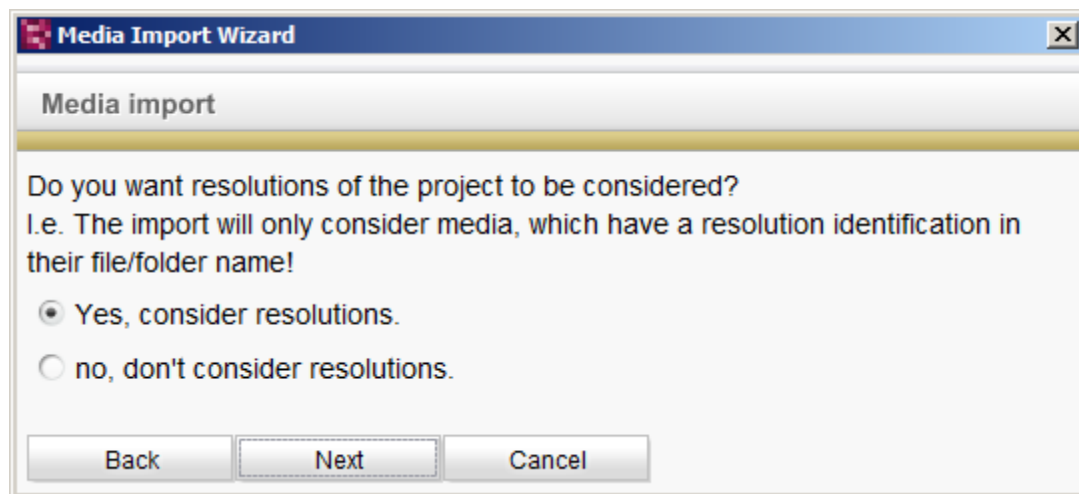


Figure 6-23: Media Import Wizard – Step 5

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, don't 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

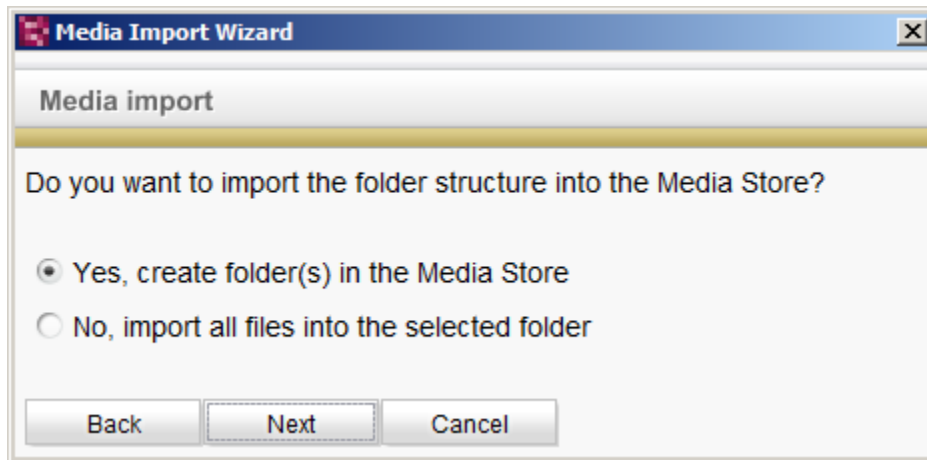


Figure 6-24: Media Import Wizard – Step 6

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

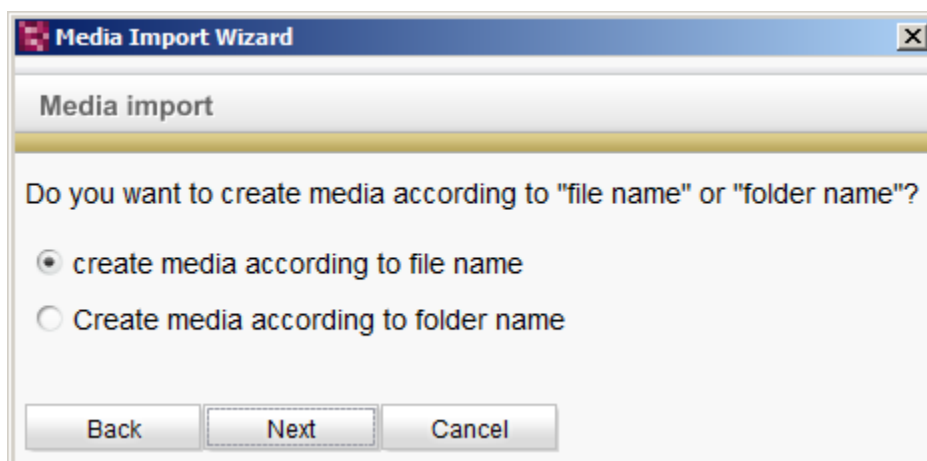


Figure 6-25: Media Import Wizard – Step 7

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 characterized 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.

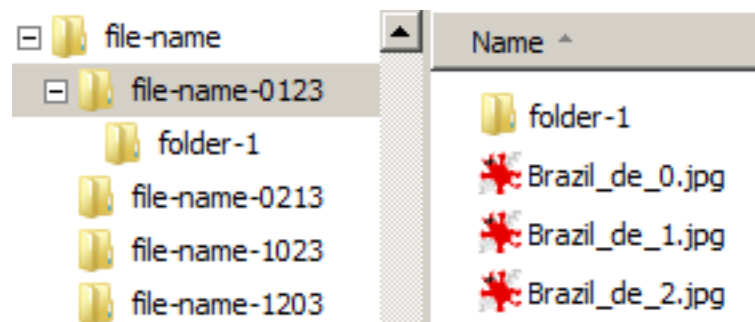


Figure 6-26: Create media via file names



The eighth step is then as follows:

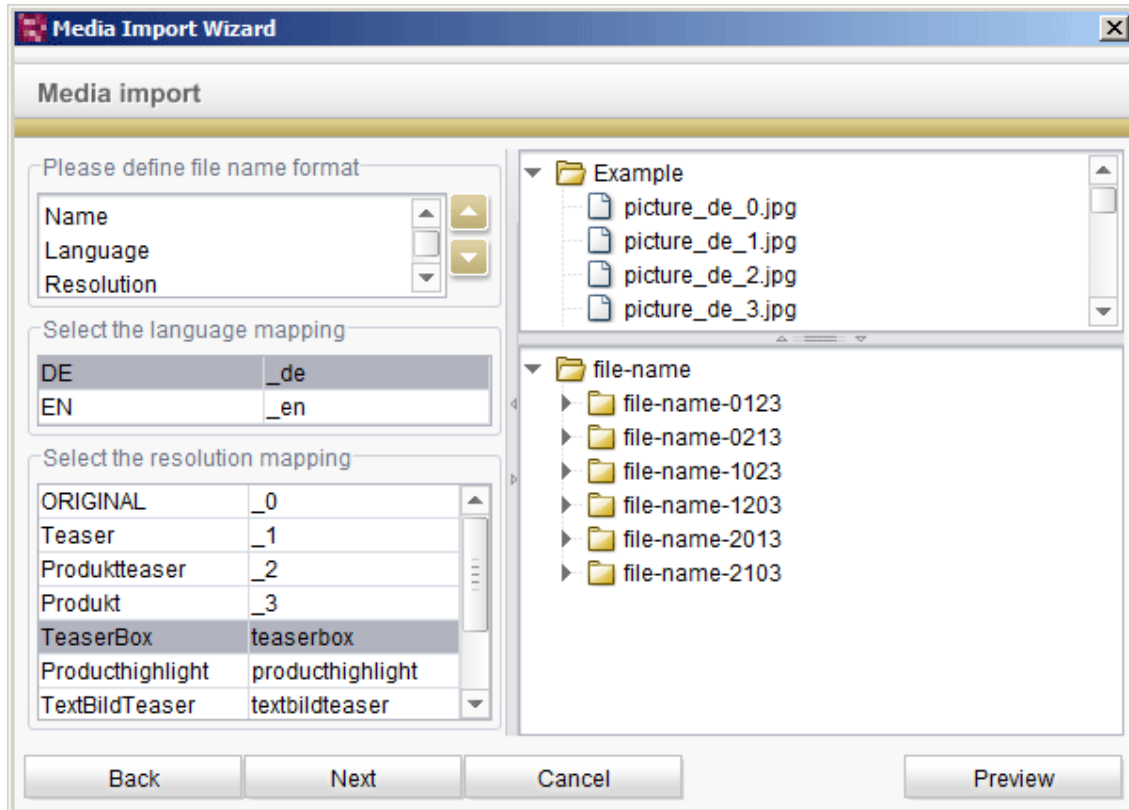


Figure 6-27: Media Import Wizard – Step 8 (Example 1)

**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.



**Example 2:**

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.

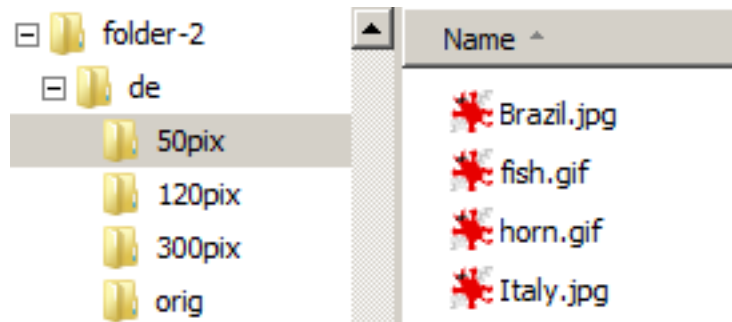


Figure 6-28: Create media via folder names

The eighth step is then as follows:

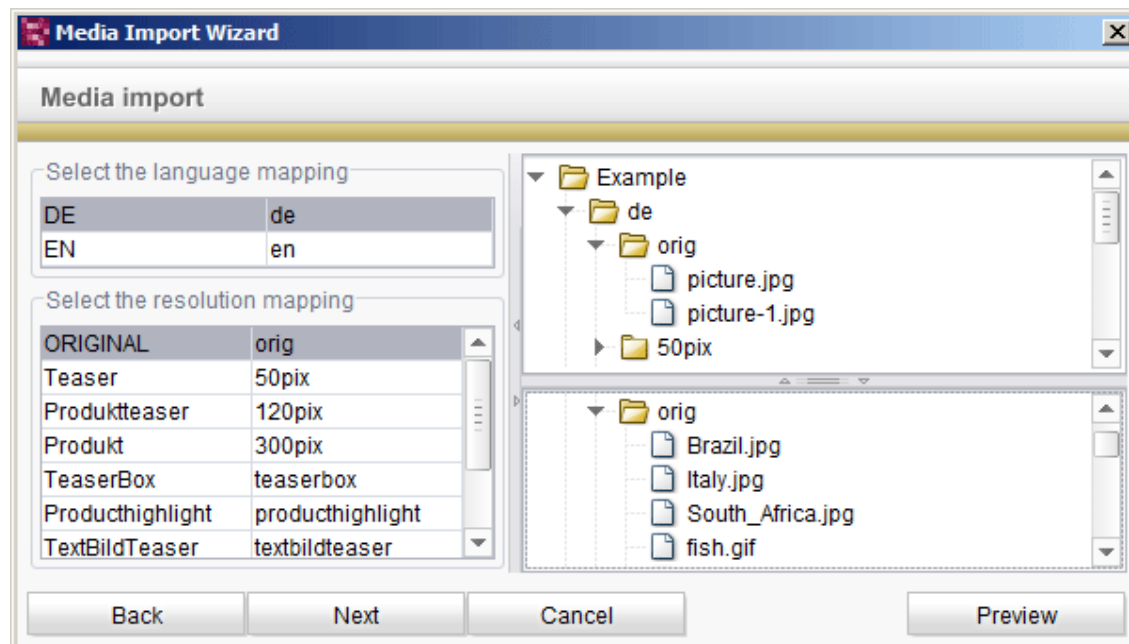


Figure 6-29: Media Import Wizard – Step 8 (Example 2)

**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 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)

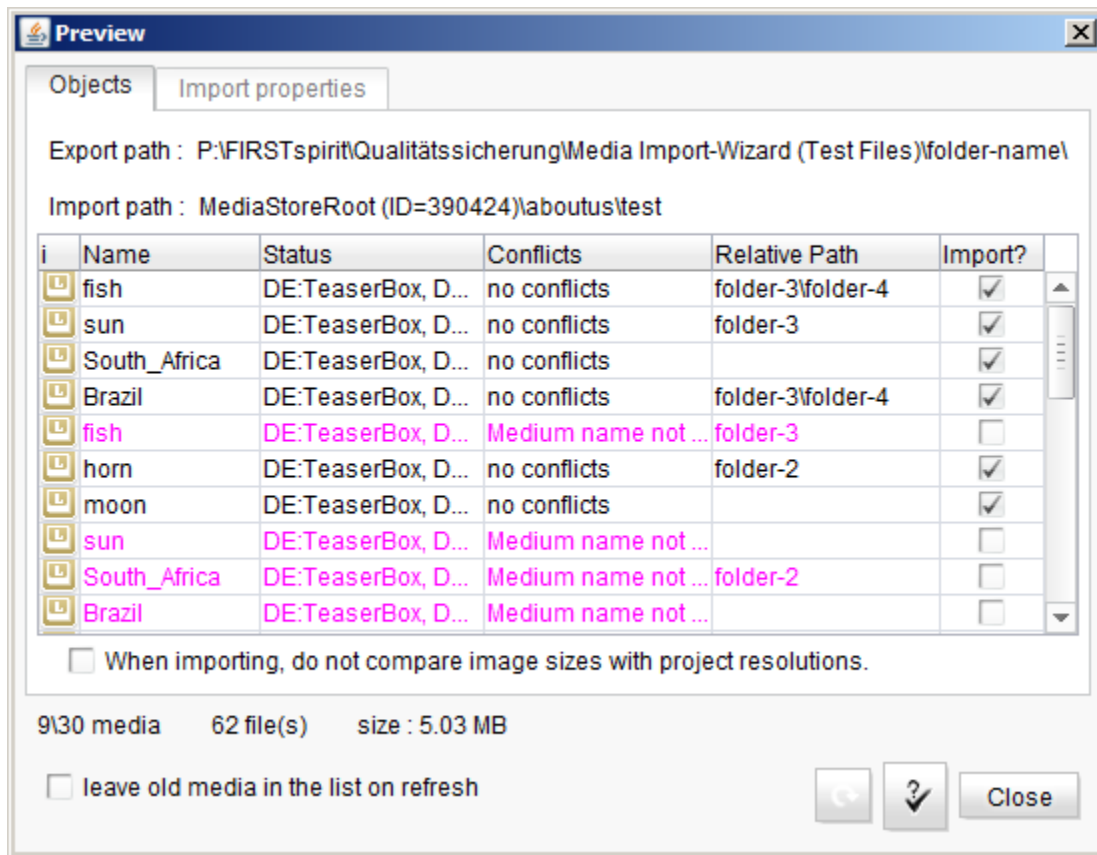


Figure 6-30: Media Import Wizard – Preview – Objects tab

Click the heading fields to display the media sorted by the values in the corresponding column.

**Name:** This is the 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:** Specifies 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 import or not. All selected media are labeled with a check mark. Click the box to set or remove a check mark.

If you right-click on a medium, a context menu opens with the following functions:

Remove from the list: The selected medium is removed from the list.

Rename: The name of the medium can be changed, for example to avoid a name being issued twice.

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.

Select all: All the media in the list are highlighted.

Invert Selection: The selections in the list are reversed. All unselected media are selected and all selected media are unselected.

Import all objects: All media without conflicts are selected for the media import, i.e. the check mark is set in the last column.

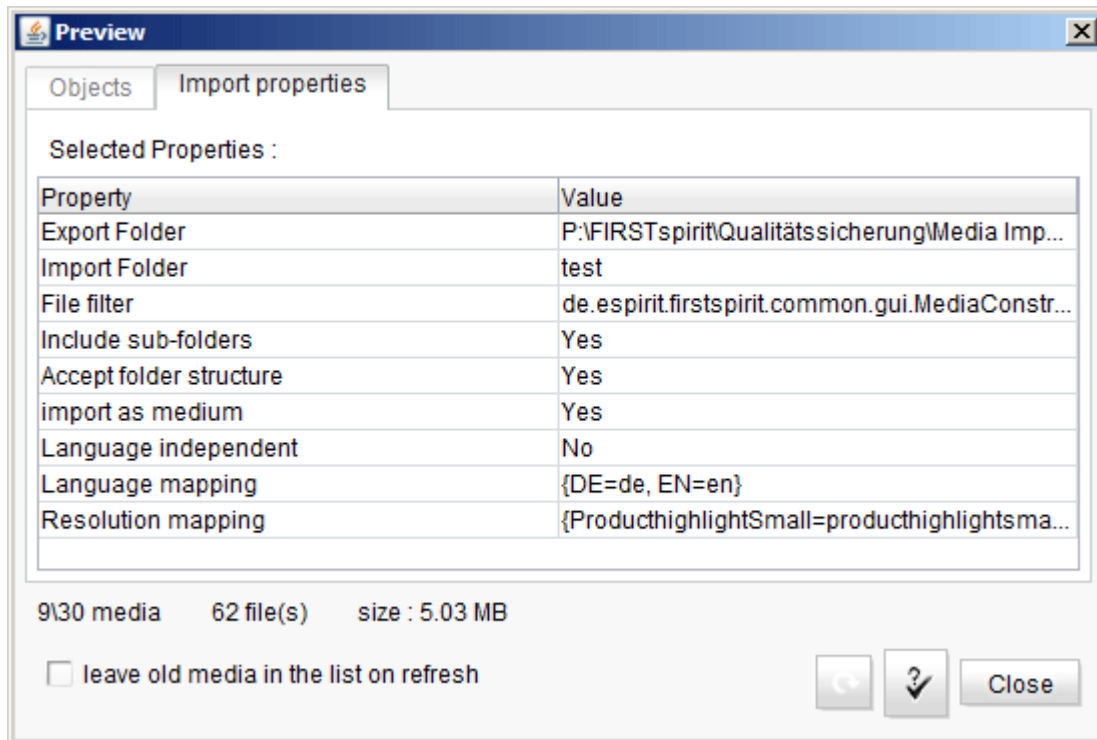
Import no objects: The import selection is undone for all media, i.e. all check marks are removed from the last column.

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 6-31: 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

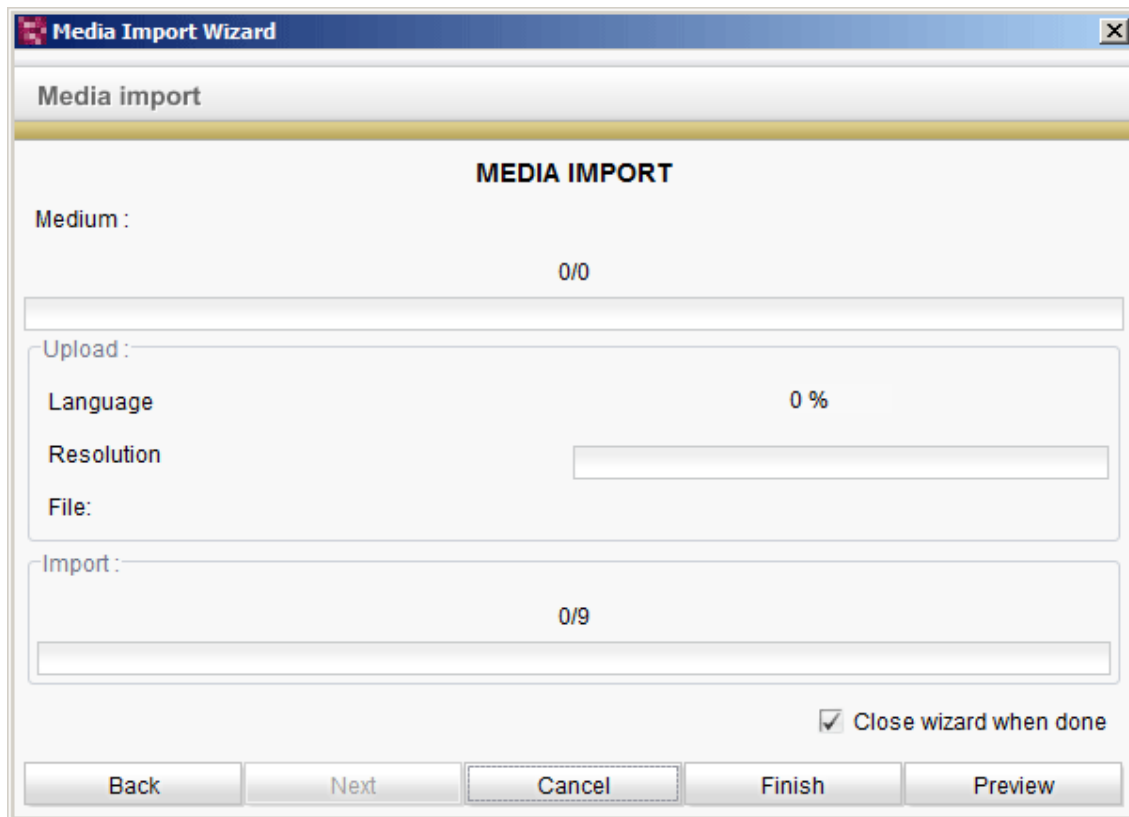


Figure 6-32: Media Import Wizard – Step 9






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

The Site Store maps 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 General Site Store context menus

The Site Store context menus are described in the following sections:

For general information on handling context menus see section 4.1 General context menus in the Page Store page 128.

#### 7.1.1 New

The "New" context menu entry can be used to insert new objects into the project.

**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. After selecting a page and confirming your selection with **OK**, another window opens in which you can assign a **language-dependent display name** for the page reference in the editing language defined for each project



administrator. Either the display names or the reference name are then displayed in the tree view, depending on the setting in the "View – Preferred display language" menu (see section 3.1.5.2, page 55). The Reference Name field is automatically filled with the value for the page used from the Page Store, but can be changed (up to the initial creation of the object). The reference name may not contain any spaces, special characters or symbols.

Next, the new page reference is included in the navigation structure. The first page inserted into a menu level 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. 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 "View – Preferred display language" menu (see section 3.1.5.2, page 55). 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.

**New – Create menu level:** 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 menu name.

**Figure 7-1: New – Create menu level**

A **language-dependent display name** can be assigned to the new menu level, 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 "View – Preferred display



language” menu (see section 3.1.5.2, page 55). 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.*



*The project administrator can define rules to automatically convert special characters in reference names 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 name"). 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 be the same as the name of the navigation level on the actual website. How to change the name is described in section 7.4.1.2 page 241.*

### 7.1.2 Edit 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.

### 7.1.3 Undo

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



#### 7.1.4 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.

#### 7.1.5 Copy

This function is used to generate a copy of the current object and store it in the (temporary) clipboard. This copy can be inserted in another position in the tree structure.

#### 7.1.6 Paste

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

#### 7.1.7 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.

#### 7.1.8 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, for example for deleting objects, can be tied to this function (see section 3.2.9.7, page 87). In this case, a dialog for starting or switching a workflow will open instead of the dialog for confirming the deleting process.*

#### 7.1.9 Display in current/new workspace

These functions allow you to choose the number of open tabs in the editing area of the FirstSpirit Client.



**Display in current workspace** opens the selected object in the central part of the screen on the active tab. **Display in new workspace** opens the selected object in a new tab.

### 7.1.10 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 section 11.10.6 page 384.

### 7.1.11 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 section 12 page 390.

### 7.1.12 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.

## 7.2 Special context menus in the Site Store

### 7.2.1 Add to Content transport feature

This license-dependent function allows you to conveniently reuse project content between projects.

A detailed description of this is given in the *FirstSpirit CorporateContent* module documentation.

### 7.2.2 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 canceled.

### 7.2.3 Set as 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.

### 7.2.4 Reference to another 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.

To carry out this function, the page reference must be in Editing mode.

### 7.2.5 Preview (page reference and document group level)

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 section [□ page 260](#)), 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 document group 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.

### 7.2.6 Show errors (page reference and document group level)

If errors occurred during the generation of the preview they can be displayed here.



### 7.2.7 Preview (release) (page reference and document group level)

This entry shows a preview of the current release status of the page or document group , i.e. the status last accepted by a person authorized to release.

### 7.2.8 Preview errors (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 the document group too, they can be displayed here.

### 7.2.9 Export (at folder and page reference level)

This function is available to administrators only.

This function can be used to export the selected folder 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 puts great strain on 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.*

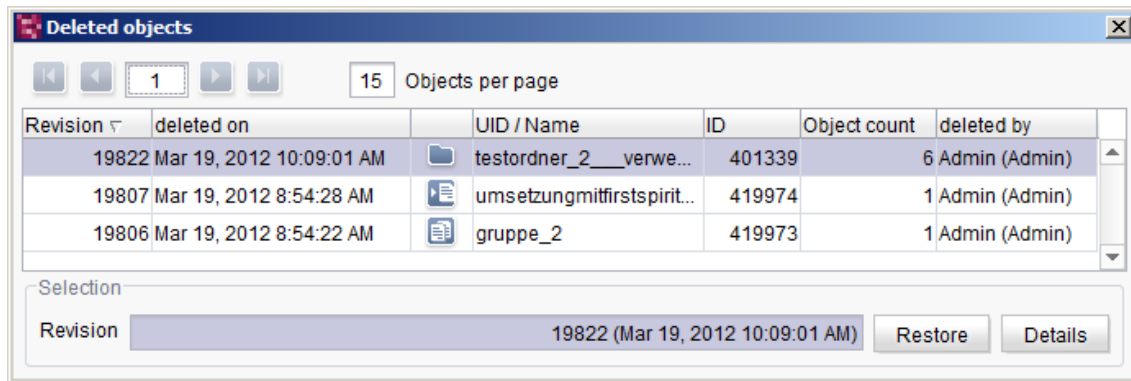
### 7.2.10 Import (at root and folder level)

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.

### 7.2.11 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.





**Figure 7-2: 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

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

**Type of object:** Object icon indicating the type of object that was deleted

**Name:** The reference name of the deleted object

**ID:** The unique ID number 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 by:** Name of the user who deleted the object.

To restore, simply select the required object and press the **Restore** button.

Restoration options and the position of the object to be restored can be determined in two further dialogs.



*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, images 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)*

## 7.3 Functions under the Extras context menu of the Site Store

### 7.3.1 Release

This function is only available for administrators.

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 section 12.4 page 399.

### 7.3.2 Delete metadata

This function is only available for administrators.

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

### 7.3.3 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 section 13.1.2, page 409.



### 7.3.4 Reset write protection

This function is available to administrators only.

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

### 7.3.5 View page (page reference level only)

This function can be used to go to the currently referenced page in the Page Store.

### 7.3.6 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.

### 7.3.7 Display properties

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. This function can also be called using the keyboard shortcut "ALT + P". The information can vary, depending on the object type.

*For further information about this function see section 4.3.11, page 142.*

### 7.3.8 Edit page (page reference level only)

This function can be used to edit the currently referenced page in the Page Store.

### 7.3.9 Cancel editing

Using this function the editing mode can be determined, without accepting changes which have been not saved yet. The status of objects that have already been released is unaffected by this action.



### 7.3.10 Change reference name

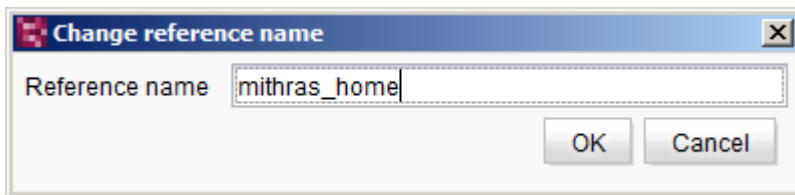
This function is only available for administrators.

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 "Display reference names in tree" checkbox and the menu item View – Preferred display language have been activated. (See section 3.1.5.2, page 55.)*

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 name" 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. an image 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 7-3: Change reference name**

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

Depending on the project administrator's settings, the menu entry may be grayed out regardless of the "Change" permission; in this case the reference name cannot be changed.

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

Depending on the project administrator's settings, the menu entry may be grayed out regardless of the "Change" permission; in this case the reference name cannot be changed.



### 7.3.11 Reset stored URLs (at menu and page reference levels)

This function is available to project administrators only.

If a path generation method which stores the generated URLs was used for generating the project (PathGeneration with "(SEO)" in generation schedules, see also *FirstSpirit Documentation for Administrators*, Chapter "Generation"), these URLs remain valid by default, even if the values of relevance for the URL generation process (depending on the path generation method selected, for example changed display or reference names, relocated pages or media in a tree, changes in datasets) should change. You can reset stored URLs by using this function so that changes in the tree structure or in display names will be reflected in the URL. See also Chapter 9.3 page 260.

This function affects elements located in the tree structure below the element on which the function was called. For example, you can reset all the stored URLs in a subtree by calling this function at the top menu level from which you want to reset the URLs. The stored URLs for all the page references and menu levels beneath this menu level are then reset.

### 7.3.12 Display dependencies

This function can be used to open the reference graph for the respective object.

For a detailed documentation of the reference graph please refer to section 11.11 page 386.

## 7.4 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 outside the navigation, therefore no navigation settings are made there.

### 7.4.1 Names tab

The website's navigation can be edited in the Names tab. Setting options are available for graphical and textual navigation.



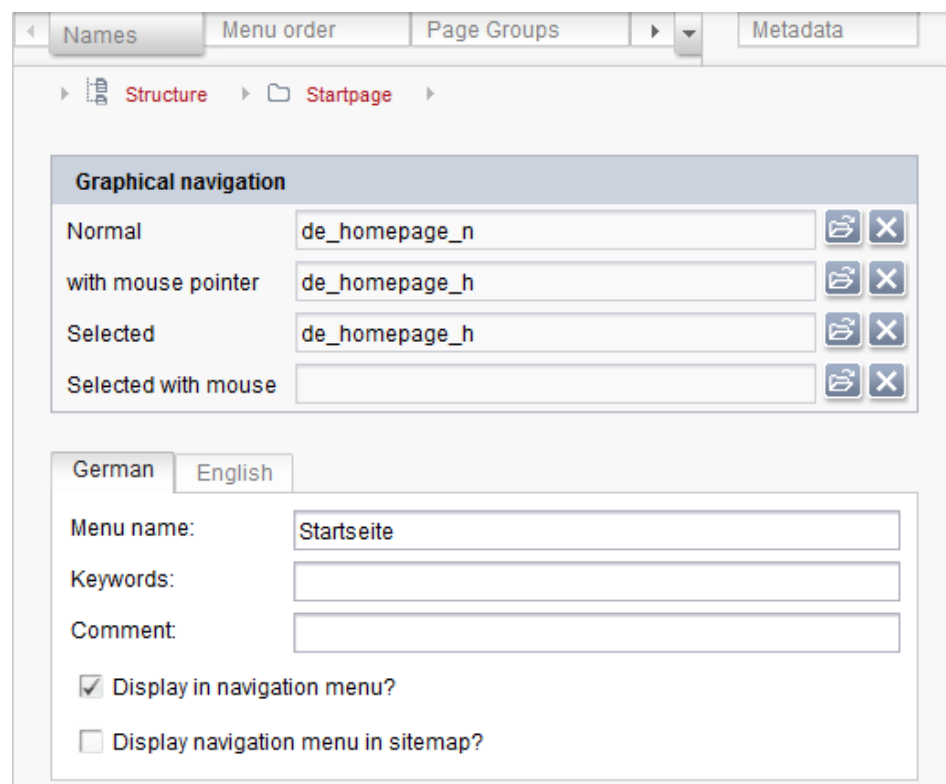



Figure 7-4: Folders view – Names tab

The Names tab is divided into three areas:

- Graphical navigation (see section 7.4.1.1, page 239)
- Display menu with text (see section 7.4.1.2, page 241)
- General details (see section 7.4.1.3, page 241)

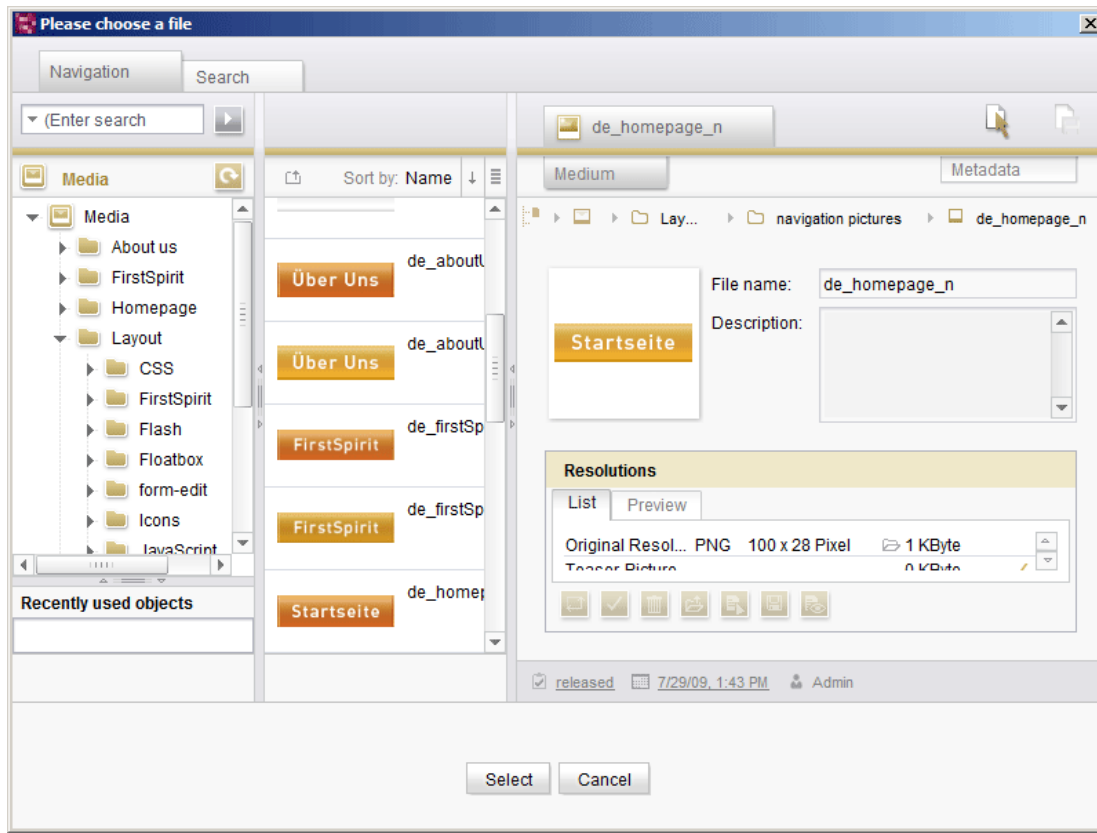
#### 7.4.1.1 Graphical navigation

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 graphics of some sort are used instead. These can be buttons with a text or real images without any text but symbols or icons clearly understood by the visitor to the website.

**Normal:** In this field you specify which graphic is displayed if the menu item is not selected. Click the  icon to open a window in which the tree structure of the Media Store is displayed.








**Figure 7-5: Select medium for graphical navigation**

The required image can be selected from the navigation structure of the Media Store. The search function in the selection dialog lets you find the desired medium quicker (see section 11.7, page 351).

After you select an image, the name of the medium you selected appears in the field. Click the Delete icon  to delete the selected graphic.

**with mouse pointer:** In this field you specify which graphic is displayed if the mouse pointer is moved over the menu item.

**Selected:** In this field you specify which graphic is displayed if the menu item is selected.

**Selected with mouse:** In this field you specify which graphic is displayed if the mouse pointer is moved over the menu item if the menu item has been selected.

These are all the states a graphic can have on a website. If you have defined an image from the Media Store for all these cases, the graphical navigation for this menu item is complete.





*As the images for the graphic are probably buttons with a text in most cases it must be ensured that language-dependent images are used, otherwise the buttons in one language would be displayed in all languages.*

#### 7.4.1.2 Display menu with text

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:** In this field you enter a text to represent this menu item within the navigation. 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 Store.

**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.

#### 7.4.1.3 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 sitemap?:** Activate this option to specify whether this page is to be listed in the sitemap too.



### 7.4.2 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.

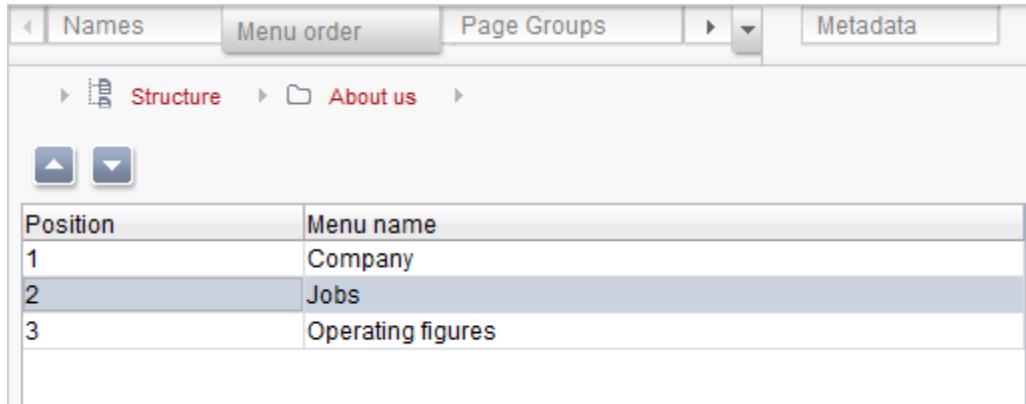




Figure 7-6: Folders view – Menu order tab

Click the Move up  or down  icons to move the selected sort order up or down by one position in the list.

### 7.4.3 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.

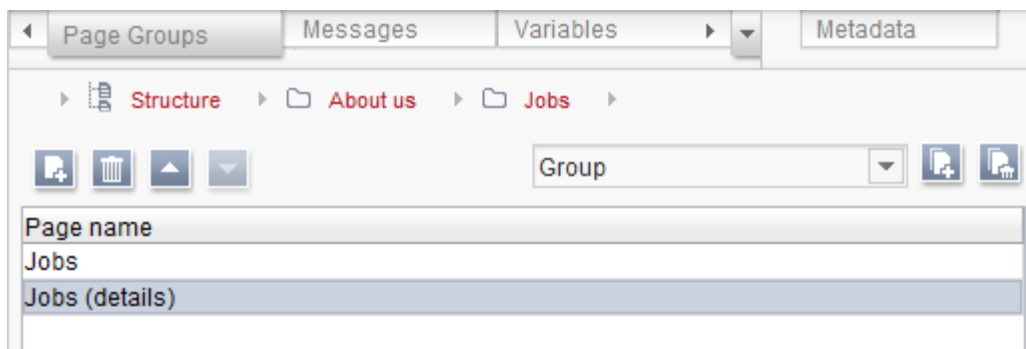



Figure 7-7: Folders view – Page groups tab

 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.



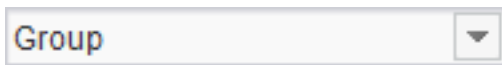
Remove this page; click this icon to remove the selected page reference from the page group.



Up; click this icon to move the selected page reference up in the list by one position.



Down; click this icon to move the selected page reference down in the list by one position.



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 check mark in front of the "Show in Page Group" entry must be removed.(See section 7.5 on page 246). 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.*

#### 7.4.4 Messages tab

The Messages tab contains a message board.

Detailed documentation on use of message boards is given in section 11.2 "The message board" page 307.

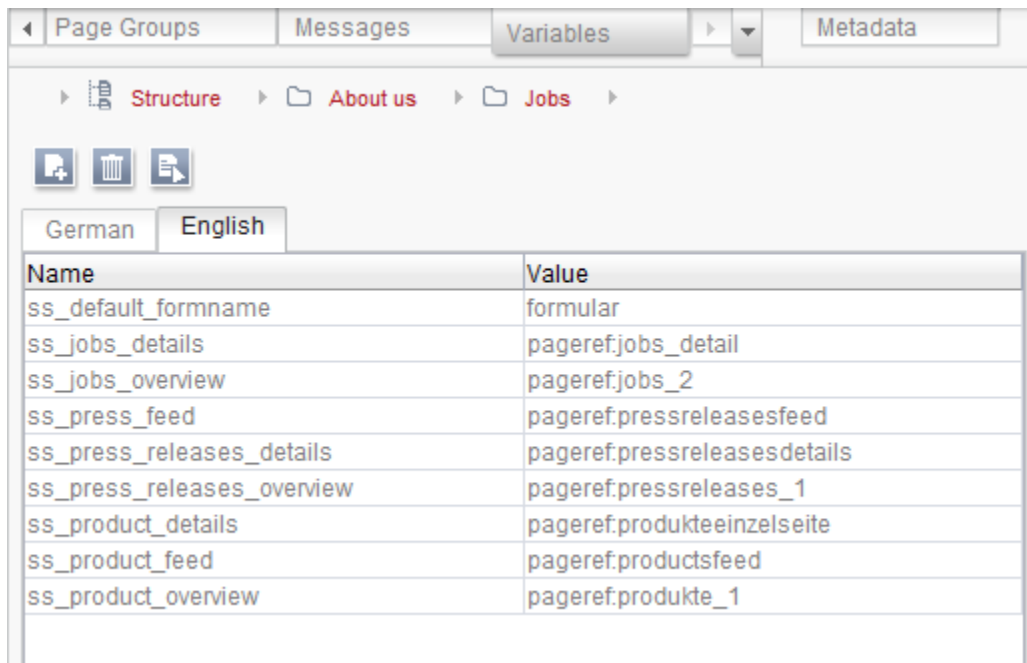


### 7.4.5 Variables tab



*This tab is visible to project administrators only.*

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 color) is to be used for the individual sub-areas of the structure, this can be done using the so-called "structure variables".



**Figure 7-8: Folders view – Variables tab**



Create a 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.

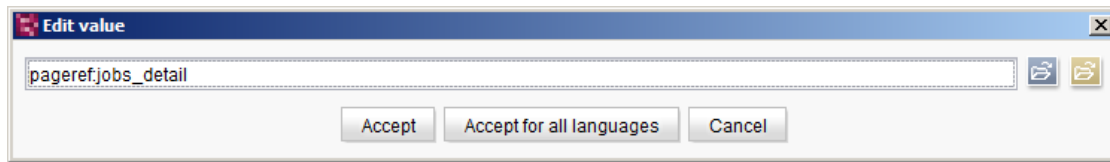


Delete Variable; click this icon to remove the selected variable.





Edit Variable; click this icon to open a window for editing the selected variable.





**Figure 7-9: 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.

Click the **Accept** button to accept the selected variable value for the current language.

Click the **Accept for all languages** button to accept the selected variable value for all the 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.*



## 7.5 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.

### 7.5.1 Languages tab

A separate tab is created for each project language.

Figure 7-10: Page reference view – Languages tab

The Languages tab is divided into four areas:

- General details (see section 7.5.1.1, page 247)
- Menu overview (Sitemap) (see section 7.5.1.2, page 247)
- Link (see section 7.5.1.3 page 248)
- Validity period (see section 7.5.1.4, page 250)



### 7.5.1.1 General details

**File name:** The file name under which the referenced page is stored on the server is entered in this field (see Figure 7-10).



*The file name may not be changed. Make sure to use a name that is 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.

### 7.5.1.2 Menu Overview (sitemap)

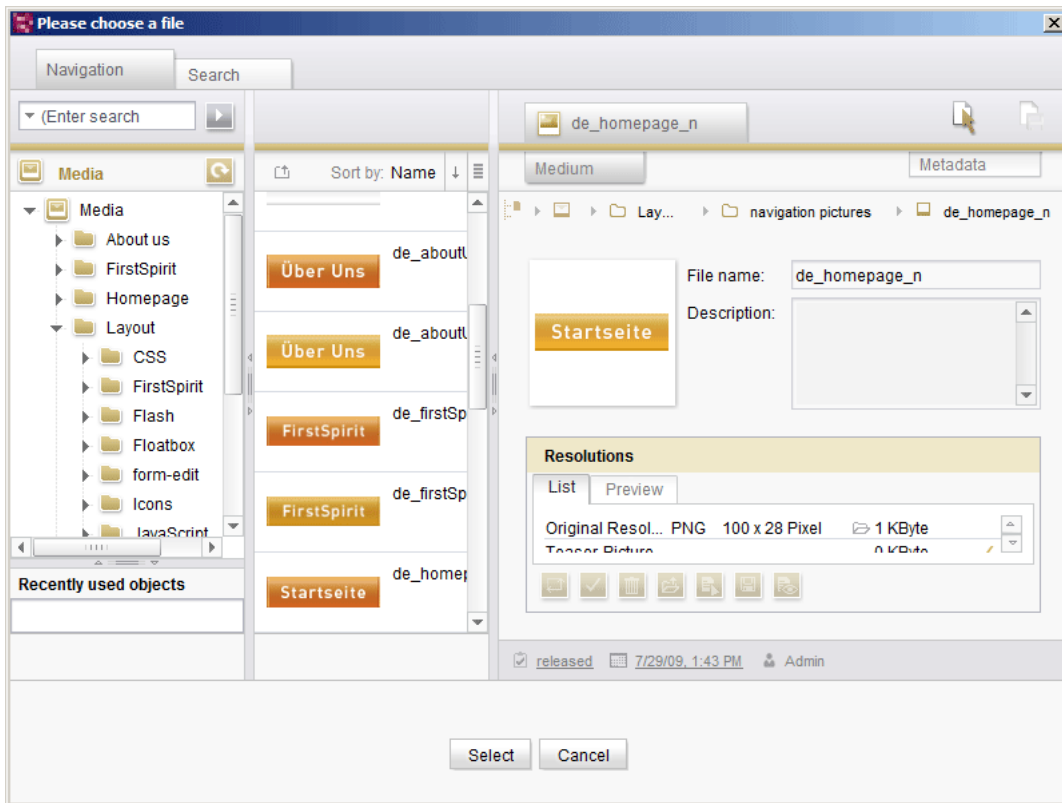
**Show in Sitemap:** If this option is activated the current menu level is displayed in a sitemap (see Figure 7-10).

**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 7-10).

**Image for sitemap:** This field can be used to select an image that 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 7-11: Select medium**

The required image can be selected from the navigation structure of the Media Store. The search function in the selection dialog lets you find the desired medium quicker (see section 11.7, page 351).

After you select an image, the name of the medium you selected appears in the field. Click the Delete icon  to delete the selected graphic (see Figure 7-10).

### 7.5.1.3 Link

One of three alternatives for the page reference can be selected in this area (see Figure 7-10).


**Internal (FirstSpirit):** This option is the default setting; all pages are generated as usual.

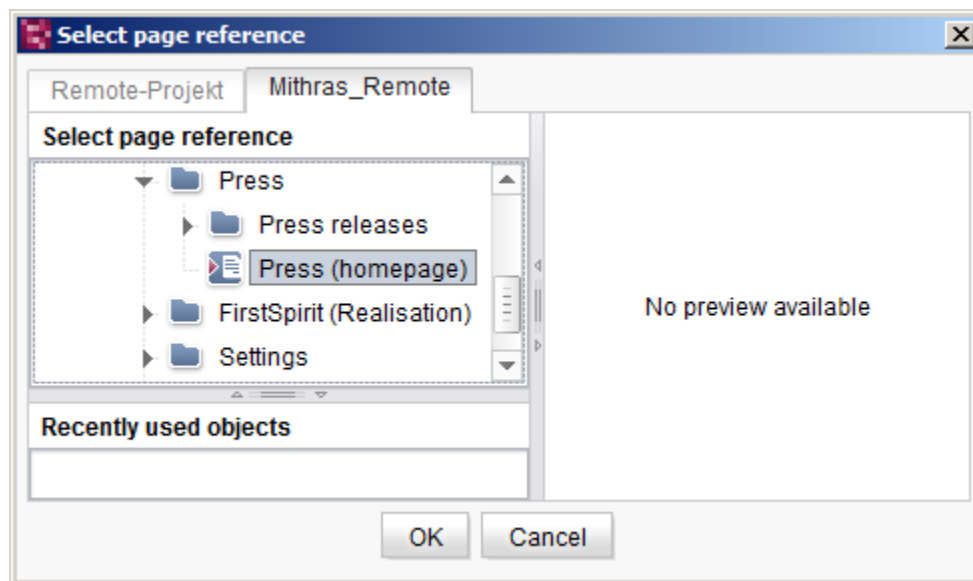
**External URL:** If a page from the project is not displayed for some reason or another, activate this option and enter a reference to the page beginning with "http://" in the field; this is the page to be displayed if someone clicks this navigation item on the website.



**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.

Click the Delete  icon to remove entered projects from this field.

Click the  icon to first select the desired project. The dialog with the Site Store of the related project opens. You can select the desired reference here.



**Figure 7-12: Select reference from the remote project**

If more than one related project is available, you can switch to them by using the individual tabs.

The selected page reference from the related project is imported as a link to the page reference of the current project. When the respective menu entry is clicked the user is directly switched to the related project page.



*Only related projects for which the project administrator has activated the "related projects" function are displayed on the tabs.*





*The "Related Projects" function is a license-dependent additional module. A detailed description of this is given in the FirstSpirit CorporateMedia module documentation.*

#### 7.5.1.4 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-clicking opens a context menu with the following entries:

**New entry:** Select this function 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 section 10.5, page 280). 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 function to remove the selected link from the list.



## 7.5.2 Content tab

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 7-13: Page reference view – Content tab**

The Content tab is divided into three areas:

- General details (see section 7.5.2.1, page 251)
- Restrictions to datasets (see section 7.5.2.2, page 252)
- Sorting (see section 7.5.2.3, page 252)

### 7.5.2.1 General details

**Number of entries per page:** Use this field to specify how many datasets 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 datasets can be output (i.e. the query is empty), use the  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 to select a column from the data source. The content of this column is then displayed in a sitemap.

#### 7.5.2.2 Limit the number of datasets

**Select a query:** The combo box can be used to select a query predefined by the project developer to limit the output of the datasets.

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.

#### 7.5.2.3 Sort order

Apart from limiting the number of datasets 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.



Add ordering by column; click this icon to use a selection box to specify another table column as the sort rule.



Remove sort order: click this icon to delete the activated sort order.



 Up; click this icon to move the selected sort rule up in the list by one position.

 Down; click this icon to move the selected sort rule down in the list by one position.

If several sort orders are specified, the top sort order 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.

## 7.6 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.



Document group

FirstSpirit (Realisation) > testordner 2 - verwendet > gruppe

File name:

Comment:

**Content**

- Implementation with FirstSpirit
- Company

**Template settings**

Start template:

End template:

Template end:

**Presentation channels**

All

html     pdf     RSS

Default channel:

**Figure 7-14: Document group view**

The possible settings for document groups are divided into three areas:

- General details (see section 7.6.1, page 255)
- Content (see section 7.6.2, page 255)
- Template settings (see section 7.6.3, page 256)
- Presentation channels (see section 7.6.4, page 256)



### 7.6.1 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).

### 7.6.2 Contents

The constituents of the document group can be defined in this area.



Click this icon to insert new elements in the document group.



Click 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.*






### 7.6.3 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 page 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 page template which forms the frame for a valid document in the selected presentation channel. The end template forms the "footer" of the document.

**Template end:** This field can be used to define the ending indicating templates that are used instead of the original templates of the integrated pages. These templates are used solely for display within the document group.

### 7.6.4 Presentation channels

**All:** If this option is activated all available presentation channels are taken into consideration in the generation.

**Selected (html, pdf, RSS):** 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 gray. 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

*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 form the basic framework for displaying content. They give the content of a website its respective 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 for example in 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 that are used for formatting so-called inline tables (see section 11.5.10, page 343).

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 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 they 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 Settings

The Global Settings are divided into four 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 **URL settings**, specific URLs can be defined for all page references in the project.

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 the 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 section 4.1, page 128.

Detailed documentation of **Special context menus** is given in section 4.2, page 134.

Detailed documentation of the functions under the **Extras context menu** is given in section 4.3, page 138.

Detailed documentation of the functions under the **Plug-ins context menu** is given in section 4.4, page 145.

Detailed documentation of **standard input elements** is given in section 10, page 273.



## 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. Similarly to the maintenance of the editorial content, the Project settings can be applied language-dependent, i.e. individually for each project language

## 9.3 URL settings



*The functions which are described in the following require in-depth knowledge about the generation function in FirstSpirit and URLs and aims for this reason rather project administrators. See also FirstSpirit Documentation for Administrators, Chapter "Generation".*

According to the specifications, URLs must not contain any Unicode characters; as a result, these structures are based on reference names in FirstSpirit and can therefore not be fully influenced by the editor.

The URL settings give editors more control over the appearance of URLs. This includes, for example, search engine optimization (SEO) or generating multilingual URLs.

The following concepts are applied here to the generation of URLs:

- **SEO URLs**

This concept allows URLs, which were previously formed monolingually in FirstSpirit from the folder structure of the project and file names of page references, to be freely defined for every node (menu levels and page references). This not only provides the option of renaming folders / directories and files, but also allows the creation of a directory structure for the web server that completely departs from the website structure and/or project. This means that this

```
http://domain.de/de/events/cebit.html    and  
http://domain.de/en/events/cebit.html
```

can be turned into this

```
http://domain.de/veranstaltungen/cebit.html    and  
http://domain.de/events/cebit.html
```



- **Short URLs**

Short URLs are brief, easily remembered, "expressive" URLs, e.g. for so-called "landing pages". The latter term stands for specifically created individual webpages where a specific topic or offer is presented in a compact manner and which are optimized for a specific target group and the subject of the page. These are often the target pages of linked adverts in other webpages. They are mostly inaccessible via the website's navigation. Short URLs are generated in addition to the "normal" URLs.

This URL

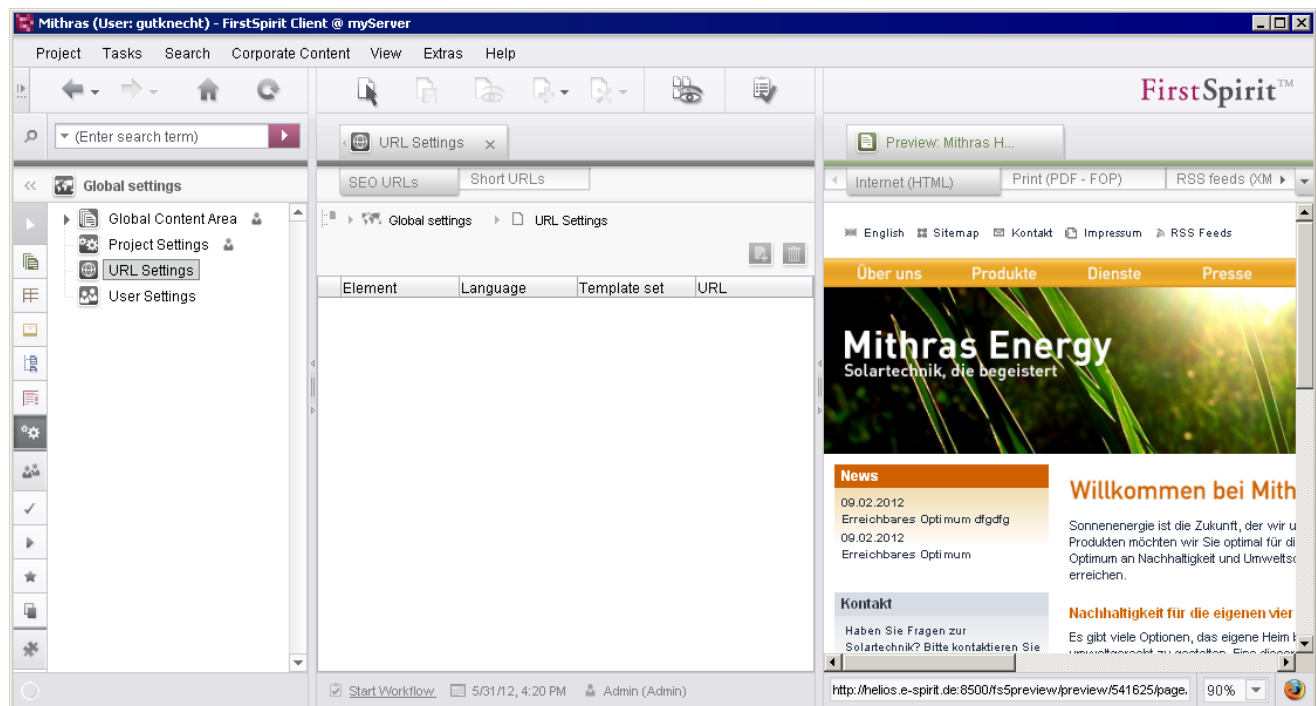
```
http://mithrasenergy.com/content/de/ueberuns/unternehmen/Unternehmen.html
```

would for example be turned into this URL

```
http://mithrasenergy.com/company
```

Every page reference can be provided with several alternative short URLs.

The JavaClient now offers the node "URL Settings" in its "Global Store" for this.



**Figure 9-1: Global Store – overview with URL settings**

The two hanging tabs

- "SEO URLs" (see Chapter 9.3.1 on page 262) and
- "Short URLs" (see Chapter 9.3.2 on page 267)



available here allow the URLs to be influenced. Both of these overviews can contain several entries.



*Project administrator rights are required for editing the tabs.*

**Element:** Shows the path and element for which a processed URL is available. This can be a page reference or a menu level.

**Language:** Shows the language the processed URL applies to

**Template set:** Shows the channel the processed URL applies to

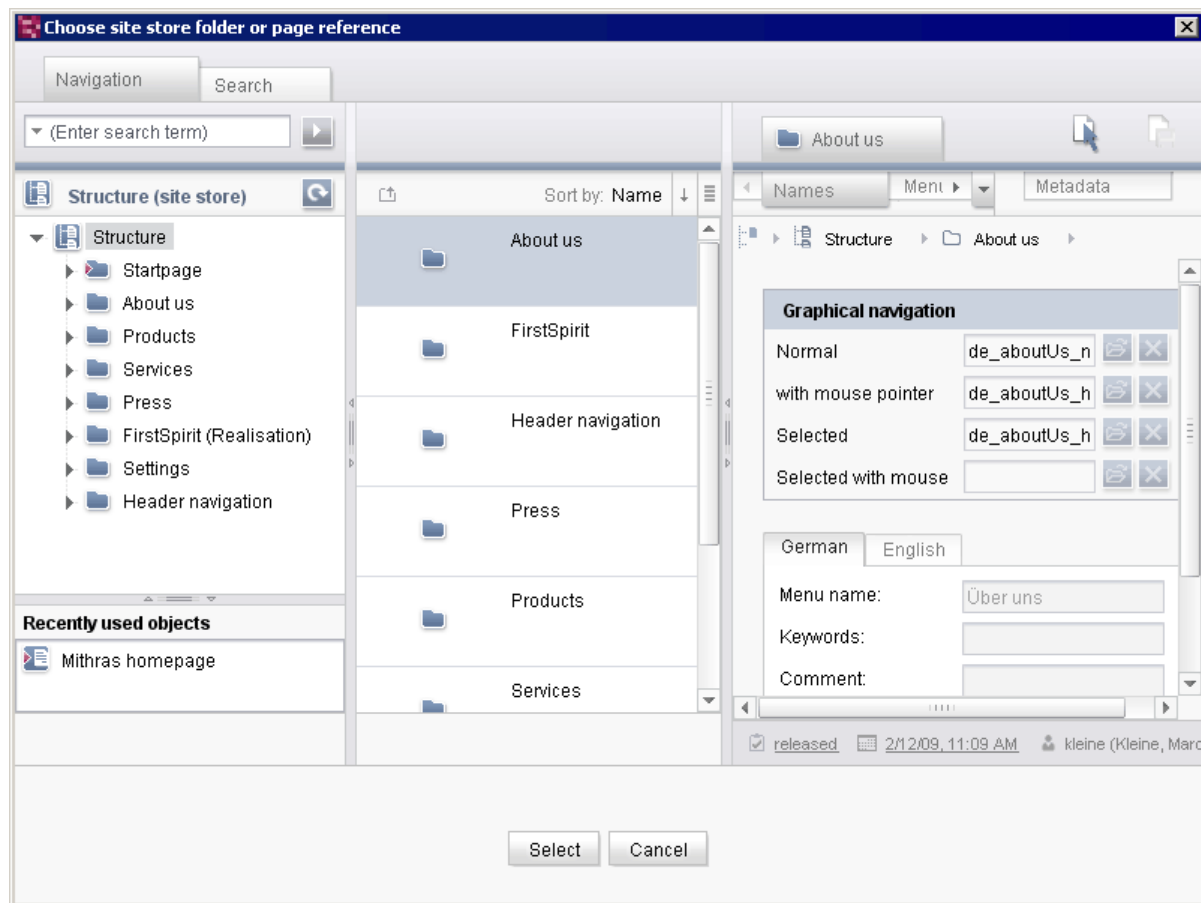
**URL:** Shows the URL created for the respective element in the generation process

### 9.3.1 "SEO URLs" tab



Add, This icon serves to select a menu level or page reference from the Site Store that a URL is to be manually specified for. The first stage opens a selection dialog:





**Figure 9-2: Editing URLs – selecting a menu level or page reference**

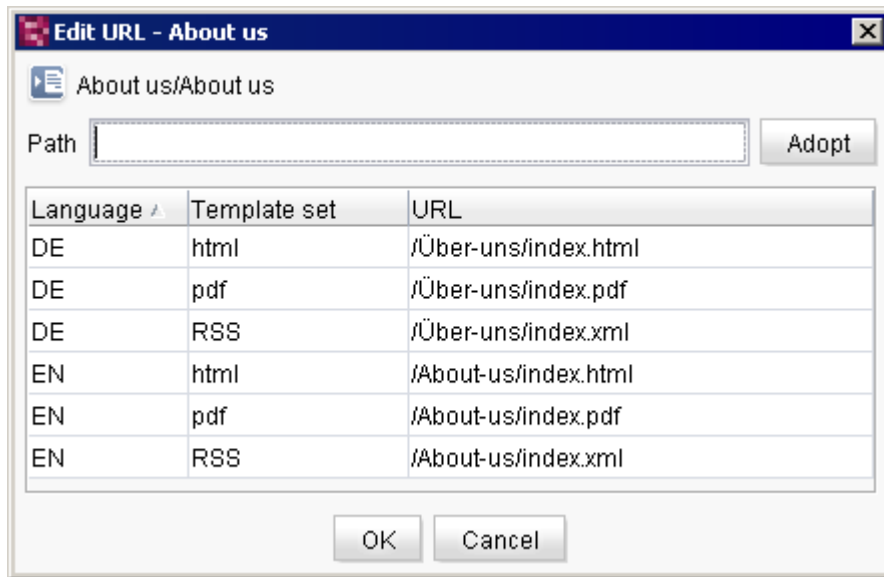
In this dialog one can select the menu levels (see "Application examples for menu level URLs") or page references (see "Application examples for page reference URLs") the URLs are to be defined for. As menu levels are converted into folders during the generation process and page references into files in a directory structure that is subsequently transferred to a web server, for example, the directory / folder path can be influenced for menu levels using the "URL settings" function, and for page references also via the file name. In the process, subdirectories are created / shown by inserted slashes ("/").

More information on working with this dialog can be found in the Chapter 11.7 page 351.

Clicking "Select" opens the following dialog:







**Figure 9-3: Editing URLs**

In this dialog, the directory paths the URLs are derived from for the selected node can be changed for the languages and template sets available in the project.

The table can be navigated by keyboard shortcuts as follows:

- **ENTER** or **down arrow key**: Go to next line / one down
- **ENTER + SHIFT** or **up arrow key**: Go to previous line / one up
- **F2**: Enter input in the "URL" field in the selected line.

If the "URL Settings" function is invoked on a menu level, only the names of paths/directories can be influenced. These changes will affect all subordinate objects that are not provided with an SEO URL definition of their own. If the function is invoked in page references, a path with a unique file name can be defined for every language and template set.

**URL:** The "URL" field in the table initially shows the currently saved URL of the node for every language and every template set. The entries can be sorted by clicking the column header. The URLs can be edited by double-clicking the fields or pressing ENTER + CTRL. Slashes at the beginning of entries are automatically inserted for both menu levels and page references.

Each URL (path plus file name) may meanwhile only be defined once within a project.

Changed entries are shown in italics. If the object concerned is a so-called multi-page (page with a dataset from a data source, "content projection"), the URL setting will always only apply to the first page.



The field can also be left blank.

Path specifications can be applied to all languages and template sets by entering them in the "URL" field above the table and adopting them for all the entries in the table by clicking "Adopt". If the "URL" field is left blank, clicking "Adopt" will delete all entries in the "URL" column. Only filled entries will be applied. If the path is only to be applied to specific entries, the corresponding entries must be selected beforehand; several entries can be selected by pressing CTRL or SHIFT at the same time. The path names can also be adjusted later on. In doing so, a slash will be automatically placed at the beginning of entries.

The entries provided here are also used for building the directory structure where the results of the generation process are filed (see *FirstSpirit Documentation for Administrators*, Chapter "Generation").

### Application examples for menu level URLs

The options available for menu levels for example include the possibility of

- assigning **other names** than the display names in the tree structure.  
This

```
../Startpage
```

could for example become this

```
../Welcome
```

by entering "/Welcome" in the "Startpage" node.

- shortening **paths**.  
This

```
../Pressemitteilungen/Presse/Mithras Energy erhält Solarpreis der Stadt  
Sonningen.html
```

could for example become this

```
../Pressemitteilungen/Mithras Energy erhält Solarpreis der Stadt  
Sonningen.html
```

by defining "/Pressemitteilungen" in the "Presse" node.



- or of adding **additional sub-directories**:

```
../Pressemitteilungen/PDF
```

```
../Pressemitteilungen/RSS
```

### Application examples for page reference URLs



*These examples relate to the "Mithras Energy" demo project, "Advanced URLs" generation mode (see FirstSpirit Documentation for Administrators, Chapter "Generation").*

An `index.*` file with the file extension of the respective template set (`*.html`, `*.pdf`, `*.xml` etc.) will be generated for page references by default.

In the case of page references

- different (file) names can be assigned by changing the file name,
- the path to a file can be shortened or expanded by removing or adding directories (with `"/`).




*The relevant file extensions must be assigned in the process.*

OK

Clicking this button applies the settings to the overview (see Figure 9-1).


If URLs (path plus file name) have been assigned several times over (other languages and/or other template sets), a corresponding message will be displayed ("The URL '...' is used repeatedly!").


If a manually entered URL is already being used for another node in the project, the duplicate URLs are shown in red in the overview. When saving or exiting the editing mode using CTRL + S or CTRL + E, respectively, or the corresponding icons in the JavaClient tool bar or the context menu option "Editing on/off", the message "Please correct duplicate URLs first." will be displayed.

The  icon will display page references and their attendant settings in the overview (Figure 9-1).

URLs that are already provided in the overview can be changed by double-clicking the respective field in the "URL" column. If an element is not yet provided in the language and/or template set in the overview the URL is to be changed for, the entry can be added for the desired node using the



 icon (see top of this Chapter).

 Delete, This icon serves to delete a line of an entry from the list and hence reset the manual URL settings for this element.

URL settings and/or changes must be saved (CTRL + S or "Save" icon in the JavaClient menu bar) or the editing mode of the node exited again (CTRL + E, corresponding icon in the JavaClient menu bar, or context menu option "Editing on/off").




*The changes made here will only be included in the generation process if a corresponding schedule is carried out. This is normally done by the project administrator (see also FirstSpirit Documentation for Administrators, Chapter "Generation").*

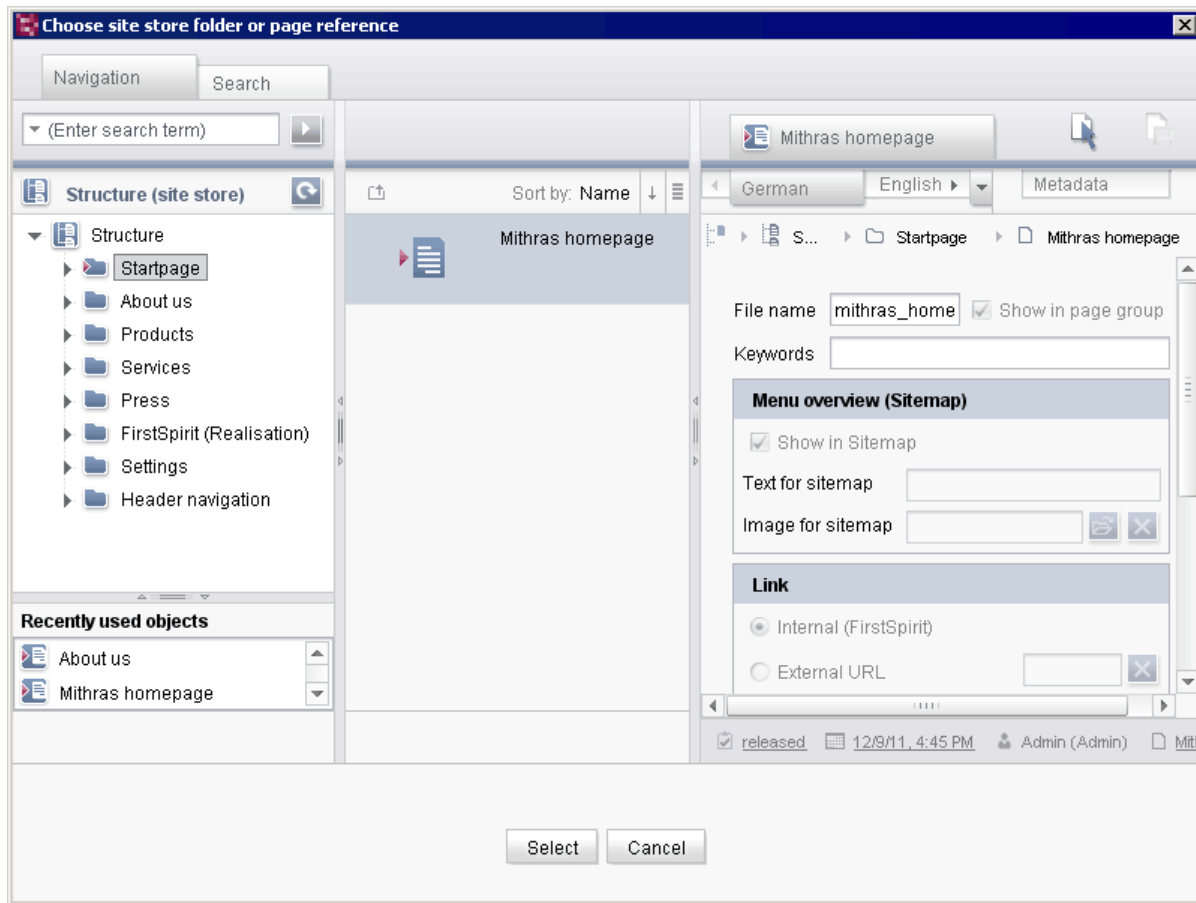


*The automatically assigned URLs that are saved in the node can be viewed in the object information (using ALT + P or context menu "Extras" / "Show properties").*

### 9.3.2 "Short URLs" tab

 Add, This icon serves to select a page reference a short URL is to be defined for from the Site Store. The first stage opens the following dialog:





**Figure 9-4: Defining short URLs – select page reference**

Several alternative short URLs can be defined for each page reference.

The definition process is the same as for SEO URLs (see Chapter 9.3.1 on page 262), except that no menu levels can be selected for short URLs.

**Application examples**

A short URL can for example be based on a **product name** or **product category** that customers and prospects are directly looking for, e.g.

```
www.mithras-energy.com/storageunit
```

or on a **specific topic** circulated by promotional materials, e.g.

```
www.mithras-energy.com/cebit
```

or



www.mithras-energy.com/offer

To create a short URL for a **product page**, for example, that is based on a content projection (page with a dataset from a data source), a page with a data source-based section must be created in the Page Store. What is required is a data base query whose result is exactly the dataset that is to be displayed in the page. This data base query can then be selected in this page's instance in the Site Store (page reference with "Content" tab). The desired short URL for this page reference can then be filed in the URL settings:

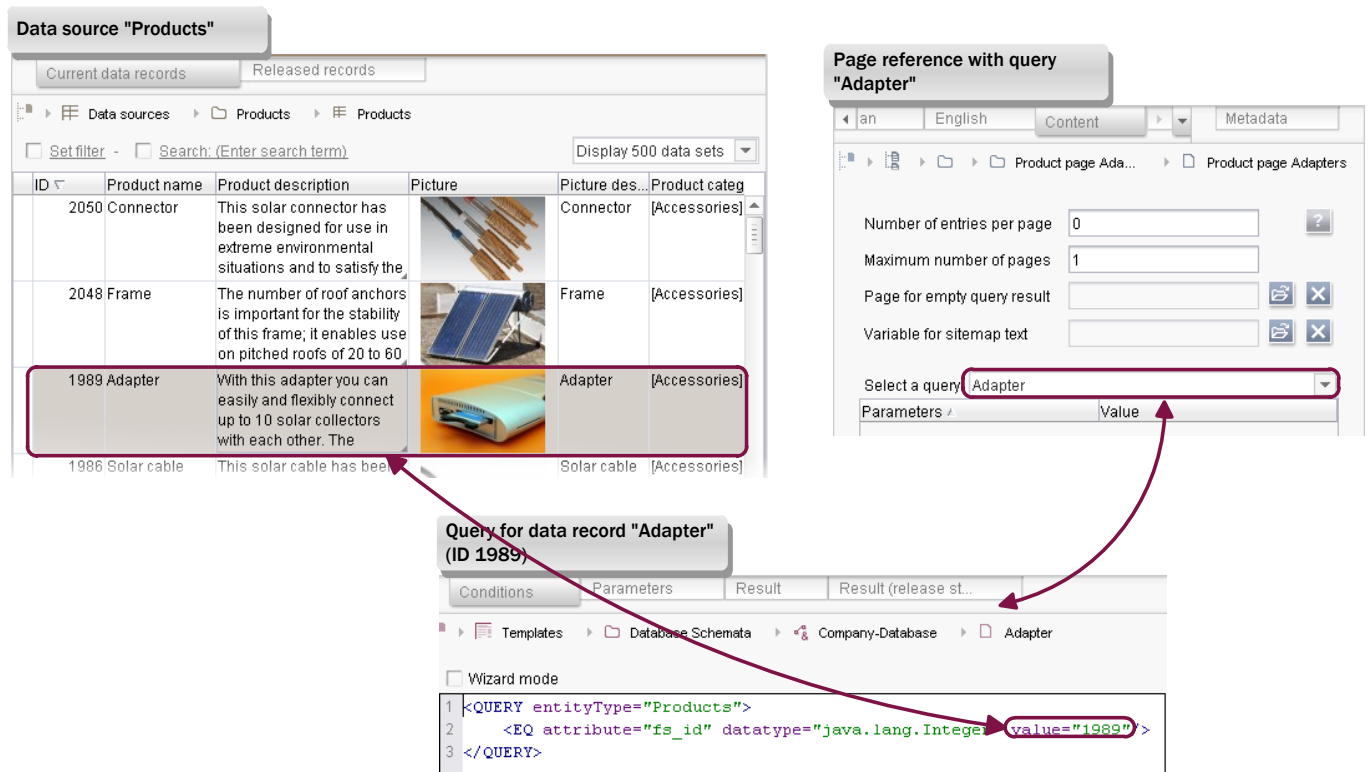


Figure 9-5: Page reference for a product page

If you want to define a Short URL for a content projection only one page may be generated. For this purpose, the following settings must be made on the tab "Content" of the page reference (see also Chapter 7.5.2 page 251):

- Number of entries per page: 0
- Maximum number of pages: 1

This icon adopts the settings in the overview (see Figure 9-1).


If URLs (path plus file name) have been assigned several times over (other languages and/or



other template sets), a corresponding message will be displayed ("The URL '...' is used repeatedly!").

If a manually entered URL is already being used for another node in the project, the duplicate URLs are shown in red in the overview. When saving or exiting the editing mode using CTRL + S or CTRL + E, respectively, or the corresponding icons in the JavaClient tool bar or the context menu option "Editing on/off", the message "Please correct duplicate URLs first." will be displayed.

Several different URLs can be defined for each page reference, however.

The  icon will display page references and their attendant settings in the overview (Figure 9-1).

URLs that are already provided in the overview can be changed by double-clicking the respective field in the "URL" column. If an element is not yet provided in the language and/or template set in the overview the URL is to be changed for, the entry can be added for the desired node using the



icon (see top of this Chapter).



Delete, This icon serves to delete a line of an entry from the list and hence reset the manual URL settings for this element.

URL settings and/or changes must be saved (CTRL + S or "Save" icon in the JavaClient menu bar) or the editing mode of the node exited again (CTRL + E, corresponding icon in the JavaClient menu bar, or context menu option "Editing on/off").



*The changes made here will only be included in the generation process if a corresponding schedule is carried out (see FirstSpirit Documentation for Administrators, Chapter "Generation").*



## 9.4 User settings

### 9.4.1 Browser tab

Several browsers for viewing preview pages can be entered in the Browser tab. This makes it possible to preview the project pages in different browsers.

These browser settings are used if the integrated preview for content is not enabled (see section 3.1.5.5, page 59) and the user-defined default browser is not to be used.

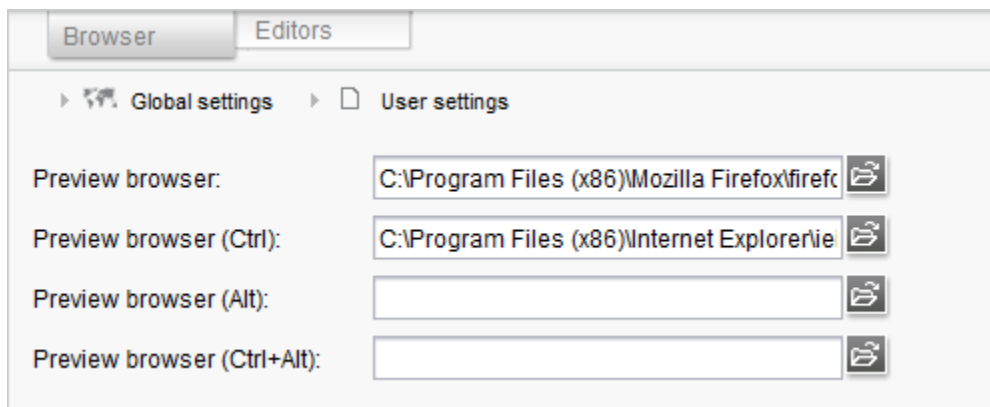



Figure 9-6: User settings view – Browser tab

**Preview browser:** Enter the default browser to be used for each preview in this field. Click the  icon to open a window with workstation's file structure. Here you can search for the exe file of the desired browser.

Further browsers can be specified in the **Preview browser (Ctrl)**, **Preview browser (Alt)** and **Preview browser (Ctrl + Alt)** rows. Open the preview with the corresponding Ctrl and/or Alt key to display the preview using the browser entered there.





## 9.4.2 Editors tab

If media from the Media Store are to be edited in an external program, you have to specify the programs in question. Suitable editors for the different file extensions can be entered in the Editor tab.

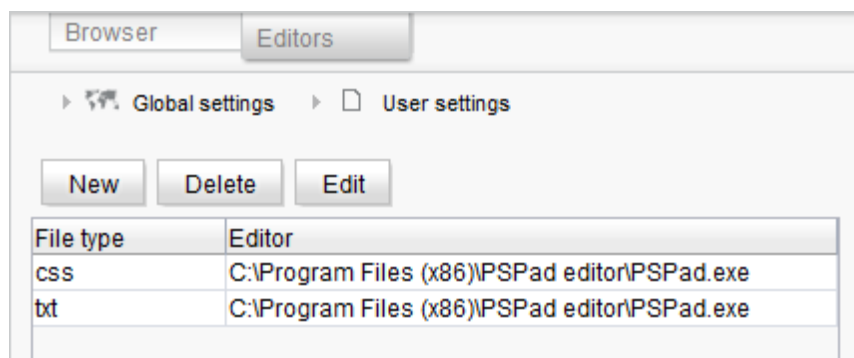


Figure 9-7: User settings view – Editor tab

**New:** Click this button to open a window in which the new file type and the corresponding editor can be specified.

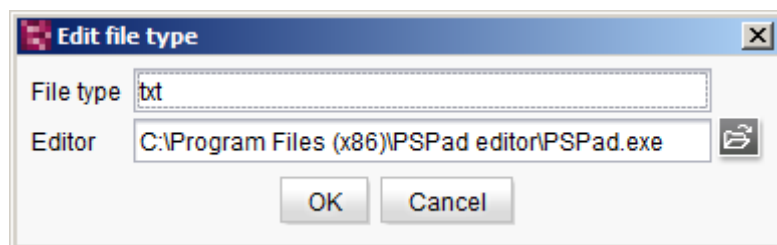



Figure 9-8: Edit file type

Click the Select  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 default input forms

FirstSpirit provides a series of input forms that editors can use to attach and manage the various types of content on the site.

The input forms can be roughly divided by function into the following groups:

### Entering texts and tables

- Single-line text (CMS\_INPUT\_TEXT) section 10.1, page 274
- Multi-line text (CMS\_INPUT\_TEXTAREA) section 10.2, page 274
- Rich text editor (CMS\_INPUT\_DOM) section 10.3, page 275
- Tables (CMS\_INPUT\_DOMTABLE) section 10.4, page 277

### Entering numbers and dates

- Numbers (CMS\_INPUT\_NUMBER) section 10.5, page 280
- Date/time selection (CMS\_INPUT\_DATE) section 10.6, page 280

### Selecting internal and external references

(images, files, pages, references)

- Reference selection (FS\_REFERENCE) section 10.7, page 282
- Link input (CMS\_INPUT\_LINK) section 10.8, page 285

### Displaying sections, datasets or links in list form

- List creation (FS\_LIST) section 10.9, page 291

### Creating and selecting datasets

- Dataset selection (FS\_DATASET) section 10.10, page 294

### Entering/selecting values and states

- Checkbox (CMS\_INPUT\_CHECKBOX) section 10.11, page 295
- Combo box (CMS\_INPUT\_COMBOBOX) section 10.12, page 296
- Radio button (CMS\_INPUT\_RADIOBUTTON) section 10.13, page 296
- Toggle (CMS\_INPUT\_TOGGLE) section 10.14, page 297
- Multiple selection list (CMS\_INPUT\_LIST) section 10.15, page 298

### Executing a function

- Button (FS\_BUTTON) section 10.16, page 299

### Specifying links to background images

- Link-sensitive graphic (CMS\_INPUT\_IMAGEMAP) section 10.17, page 300





The template developer has the option of specifying default values and invalid entries for some input forms. The effects of editing the input forms are described in section 10.18, page 302.

## 10.1 Single-line text

This input form is intended for single-line text entries, such as headings. Formatting (bold, italics, etc.) cannot be selected here, they are uniformly specified via the template development.

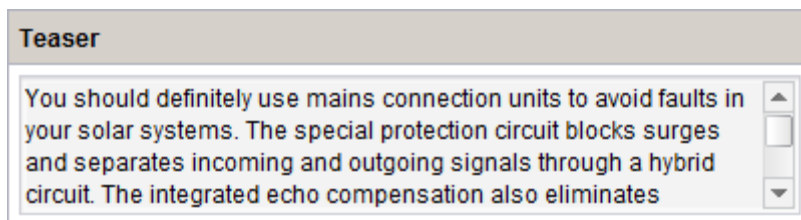
A screenshot of a single-line text input form. The form has a light green header bar with the text "Headline". Below the header is a white text box with a thin border containing the text "Welcome to Mithras Energy".

**Figure 10-1: Input form – Single-line text**



If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)

## 10.2 Multi-line text

This input form is used for larger text entries that are not to be formatted by the editor.

A screenshot of a multi-line text input form. The form has a light brown header bar with the text "Teaser". Below the header is a white text box with a thin border containing several lines of text: "You should definitely use mains connection units to avoid faults in your solar systems. The special protection circuit blocks surges and separates incoming and outgoing signals through a hybrid circuit. The integrated echo compensation also eliminates". On the right side of the text box, there are three small icons: a square with a document icon, a square with a document icon, and a square with a document icon.

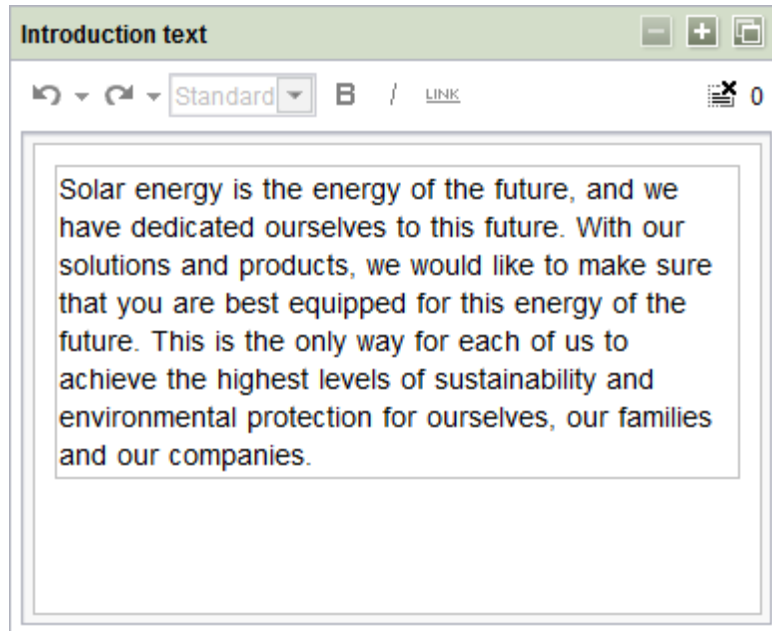
**Figure 10-2: Input form – Multi-line text**

If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)





### 10.3 Rich text editor


The rich text editor is intended for larger text entries with formatting and link specification. The functions and icons available to the editor in the specific rich text editor depend on how the template developer configured it.





**Figure 10-3: Input form – Rich text editor**

If the project developer has defined a preset default value for the input form, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)

Depending on the project's configuration, a spelling check can be carried out within the input form (see section 11.5.3, page 324).


 Open in separate window; click this icon to open the Editor window in full screen size for being able to edit large amounts of texts comfortably. The window is closed by clicking the "x" icon in the top right-hand corner or by switching to the View mode. The contents of the large window are then automatically copied into the small window.


 Predefined formatting which is to apply to the whole section can be selected here. The available formatting options are defined by the project developer.


 Predefined formatting which is to apply to the selected text only can be selected here. The available formatting options are defined in the Format templates 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.

 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. ( )

 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 italicized selected text. ( )

 Insert Link; use this icon to insert a link in the cursor's current position. The procedure for setting a reference is analogous to the description of the "Link Input" input component. (See section 10.8, page 285) ( )

 Insert List; use this icon to insert a list in the current section.




*As the rich text editor is a very complex input form, detailed documentation on working with the rich text editor input form is provided again in section 11.5, page 322.*



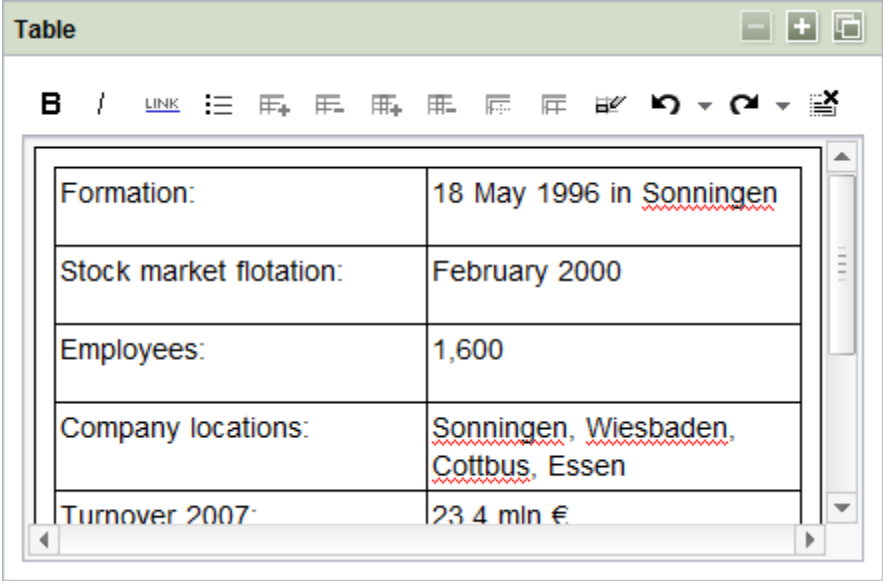
*Integration of tables into continuous text is provided by the so-called "inline tables". For more information see section 11.5.10, page 343.*





According to the settings of the template developer the functionality of the license-dependent module FirstSpirit OfficeConnect can be available. 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 OfficeConnect".



## 10.4 Tables




Formation:	18 May 1996 in <u>Sonningen</u>
Stock market flotation:	February 2000
Employees:	1,600
Company locations:	<u>Sonningen</u> , <u>Wiesbaden</u> , <u>Cottbus</u> , <u>Essen</u>
Turnover 2007:	23.4 mln €

**Figure 10-4: Input form – Table**

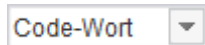
This input form can be used to enter larger text inputs with simple formatting and link inputs directly in the table view.

If the project developer has defined a preset default value for the input form, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)




Depending on the project's configuration, a spelling check can be carried out within the input form (see section 11.5.3, page 324).




 Open in separate window; click this icon to open the Editor window in full screen size for being able to edit large amounts of texts comfortably. The window is closed by clicking the "x" icon in the top right-hand corner or by switching to the View mode. The contents of the large window are then automatically copied into the small window.









Predefined formatting which is to apply to the selected text only can be selected here. The available formatting options are defined in the Format templates 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 italicized selected text. ( )


 LINK Insert Link; use this icon to insert a link in the cursor's current position. The procedure for setting a reference is analogous to the description of the "Link Input" input component. (See section 10.8, page 285.) ( )


 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 to the right of 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 more detailed description of the cell properties, see section 11.6.4, page 350.)


 Import; click this icon to open a window for importing text (.txt) or CSV files (.csv) whose content is transferred directly into a table (see Figure 10-5). Each line of the text/CSV file is written into a table row, columns can be generated by separators. **Important:** An import deletes

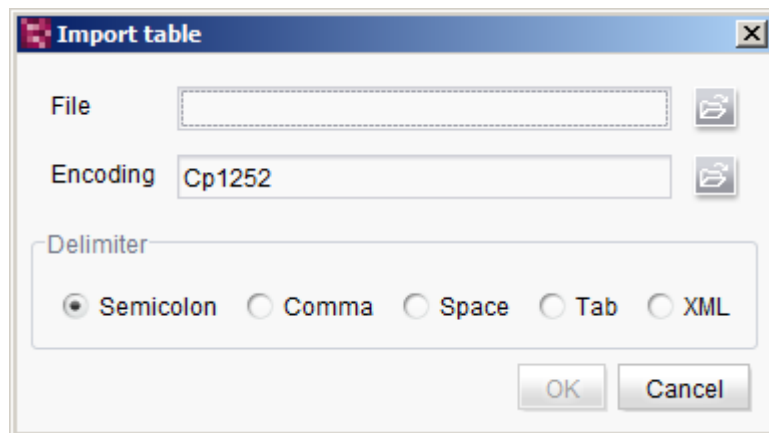


all the table content to date and replaces it with the imported content.

 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.


 Show/hide lines; use this icon to hide the table frame. Click the icon again to display the frame once more.



**Figure 10-5: DOM table input component: Import table**

In this window you can specify the details of the file to be imported.

**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 converted.

**Separator:** The separator which separates the content of adjacent cells can be defined here.



*As a table is a very complex input form, detailed documentation on working with the table input form is provided again in section 11.6, page 348.*









Additional formatting options and the integration of tables into continuous text is provided by the so-called "inline tables". For more information see section 11.5.10, page 343.



According to the settings of the template developer the functionality of the license-dependent module FirstSpirit OfficeConnect can be available. 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 OfficeConnect".

## 10.5 Numbers

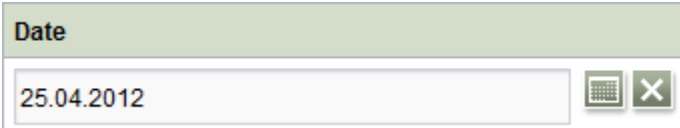


The screenshot shows a form with a title bar "Number of press releases (1-5)" and a small document icon on the right. Below the title bar is a text input field containing the number "2".

Figure 10-6: Input form – Numbers



This input form 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.

## 10.6 Date/time selection

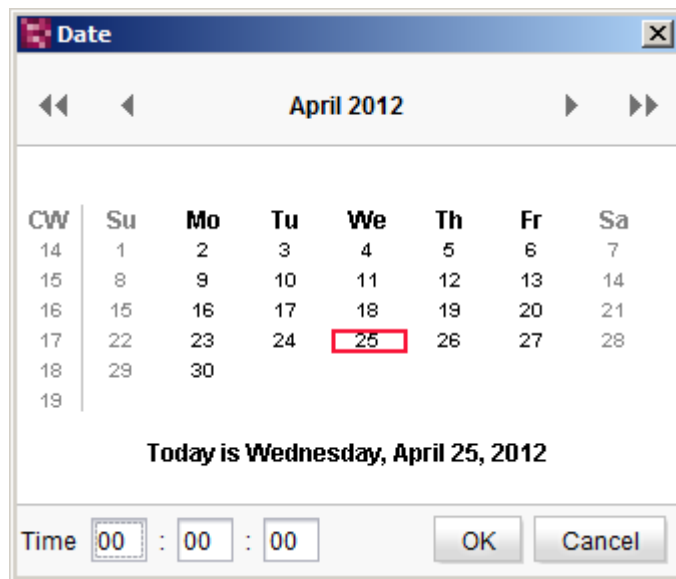


The screenshot shows a form with a title bar "Date". Below the title bar is a text input field containing the date "25.04.2012". To the right of the input field are two icons: a calendar icon and a close icon (X).

Figure 10-7: Input form – Date/time selection

The date/time selection can be used to select a date and/or a time. Click the  icon to open a window in which the date and/or time can be selected (see Figure 10-8). Click the  icon to remove the selected date and/or time from the input component.







**Figure 10-8: Select date**

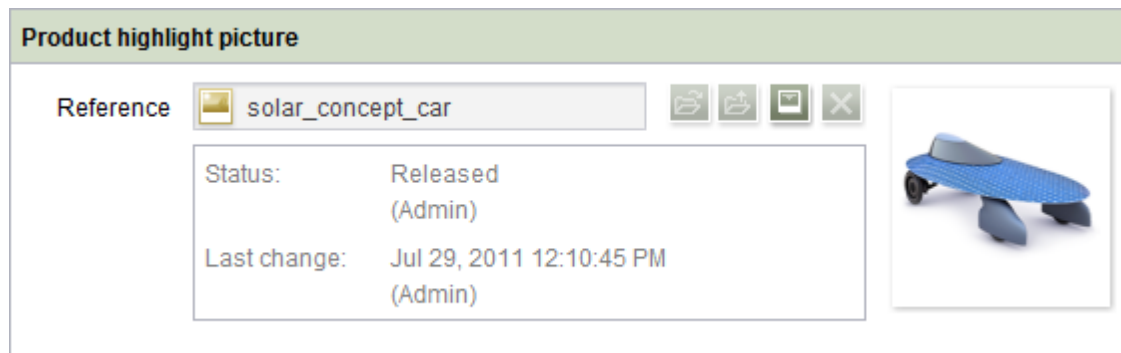
A new date can be selected within the window which is then automatically transferred into the input form in the format which has been specified by the template developer. When the dialog box is opened the currently entered date is displayed as a default (red border). The user can click the calendar view 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. Regional and national public holidays are highlighted in the calendar view in different colors. A time can be entered in the lower area of this window.

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 "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)



## 10.7 Reference selection





The screenshot shows a web interface titled "Product highlight picture". It features a text input field containing "solar\_concept\_car" with a small image icon to its left. To the right of the input field are four icons: a folder, a document, a window, and a close button. Below the input field is a table with the following information:

Status:	Released (Admin)
Last change:	Jul 29, 2011 12:10:45 PM (Admin)


To the right of the table is a small image of a blue and grey solar concept car.

**Figure 10-9: Input form – Reference selection**

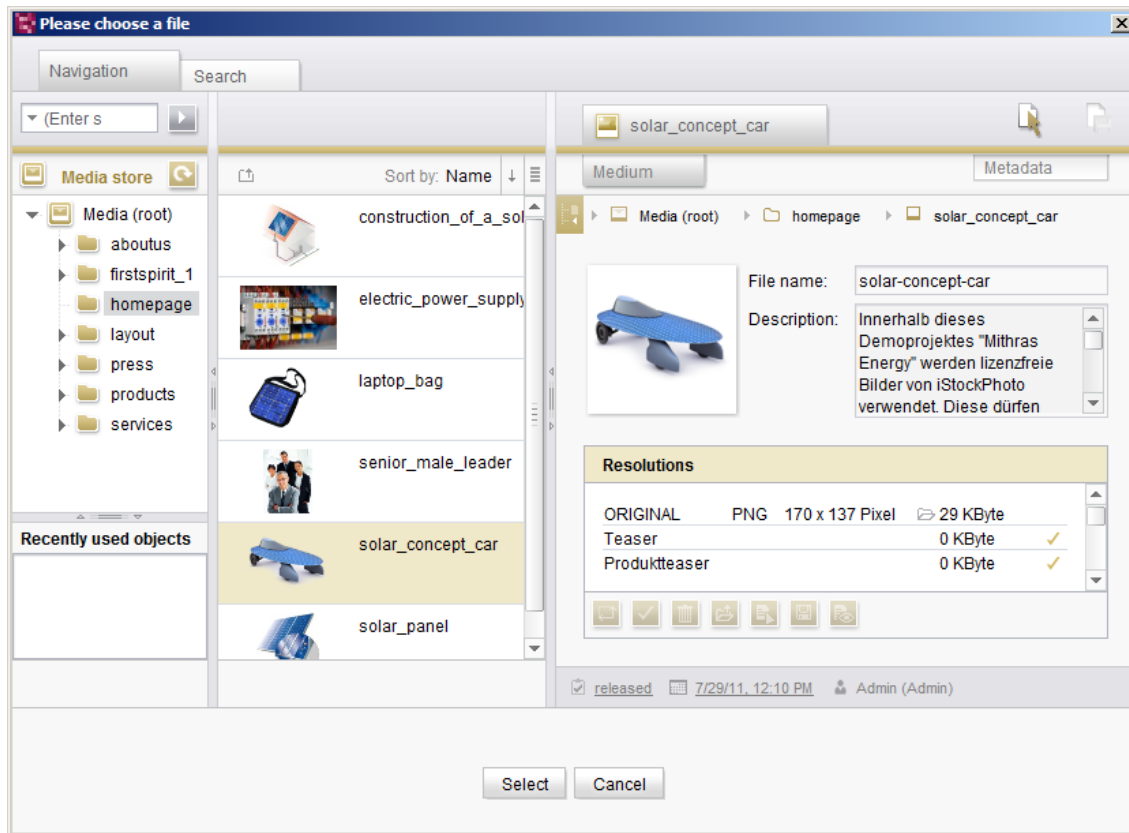
The reference selection is used to include a reference of your choice. Depending on the project developer's predefined settings, any object type can be selected. References to media (images and files), to page references (Site Store) and each other store and, e.g. to other folders.

If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)

**Reference:** This field 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:



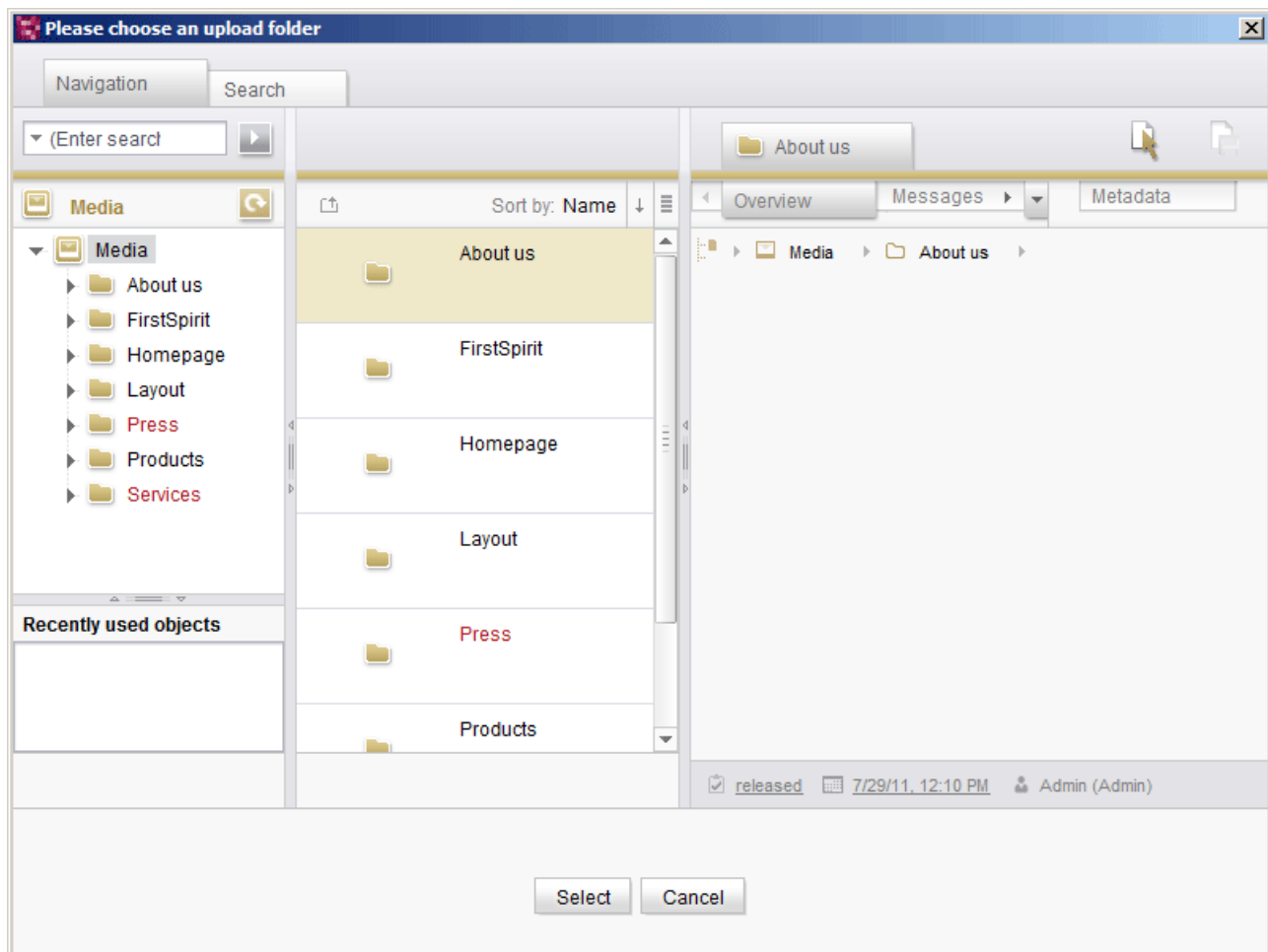


**Figure 10-10: Select reference (here: Media Store)**

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, see also section 11.7, page 351.

 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. A window for selecting the desired medium from the file system of the workstation computer will open. After confirming the selection via the button **Open** another window for selecting the desired upload folder in the Media Stores will open.





**Figure 10-11: Choose upload folder**

A language-independent medium will then be created in the selected folder in the Media Store and referenced directly in the input form.



*Remote projects are a license-dependent additional module. A detailed description of this is given in the "FirstSpirit CorporateMedia" module documentation.*



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.



## 10.8 Link input

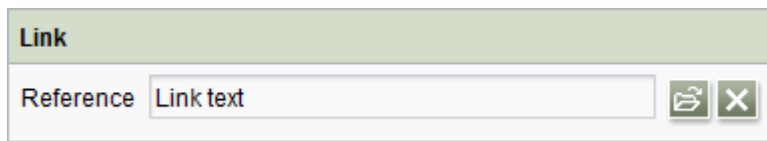



Figure 10-12: Input form – Link input

**Reference:** Click the  icon to create a link. A window opens in which the link type must be selected.

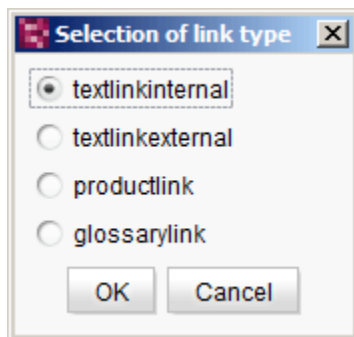




Figure 10-13: Link type selection

After selecting the link type, an input window opens whose content depends on the link type and the input components used, for example for link texts, comments or link images.

If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)



*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.*



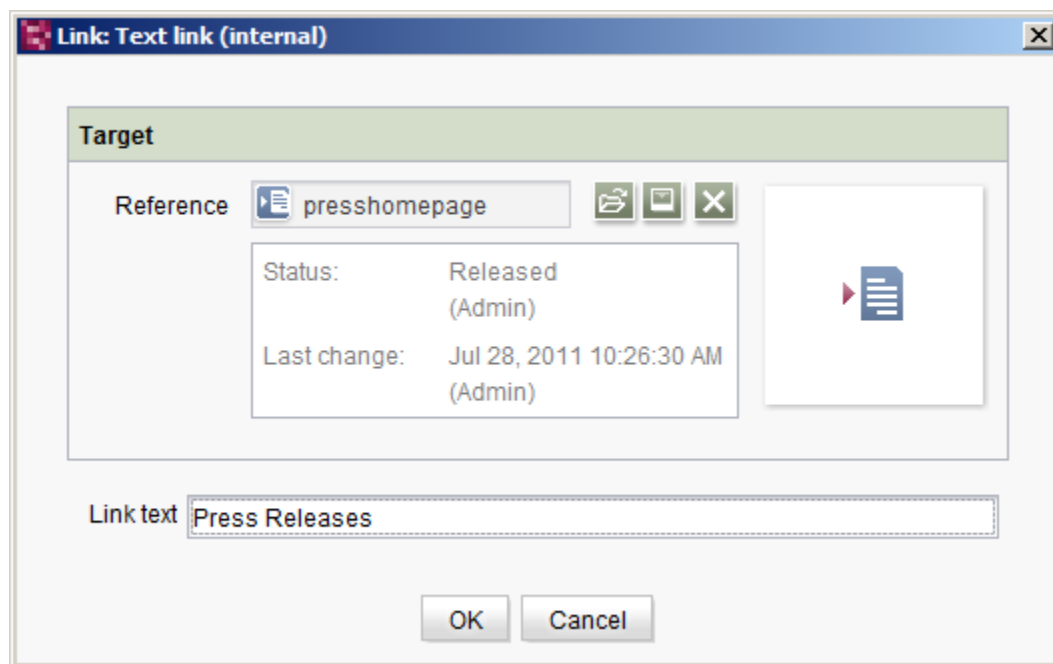
*The project developer determines which of the respective input fields are used for the link. It is therefore possible that individual inputs do not have any effect or errors occur in the display if inputs are missing.*





*If the template developer modified the respective template in retrospect, content which was entered by means of the template's input fields can be displayed in the preview in a state which is out of date in rare cases. The display can be updated by saving the content anew.*


### 10.8.1 Internal Link



**Figure 10-14: Internal Link**

This type of link is a link to another page in the same FirstSpirit project.

**Reference:** This field displays a selected reference with object symbol, release status and the time and originator of the last change.

Click the  icon to select a page reference to be linked to from the Site Store.



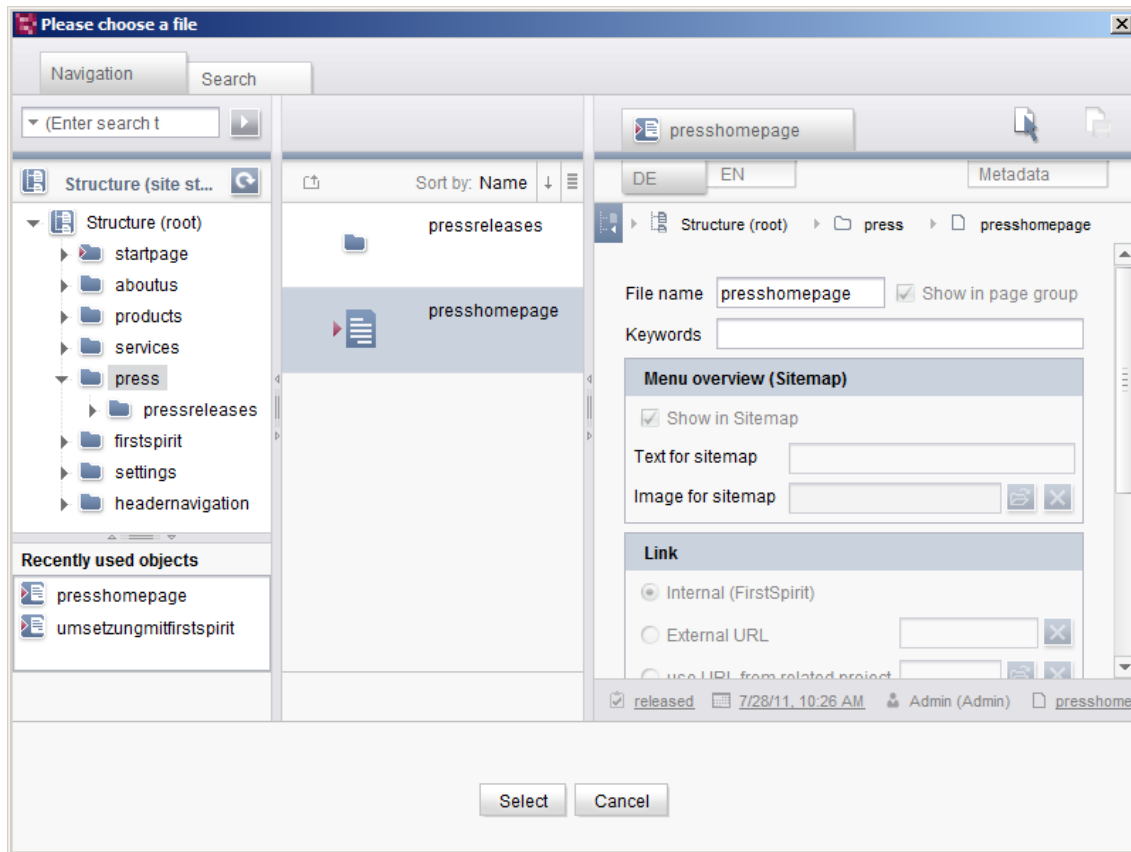


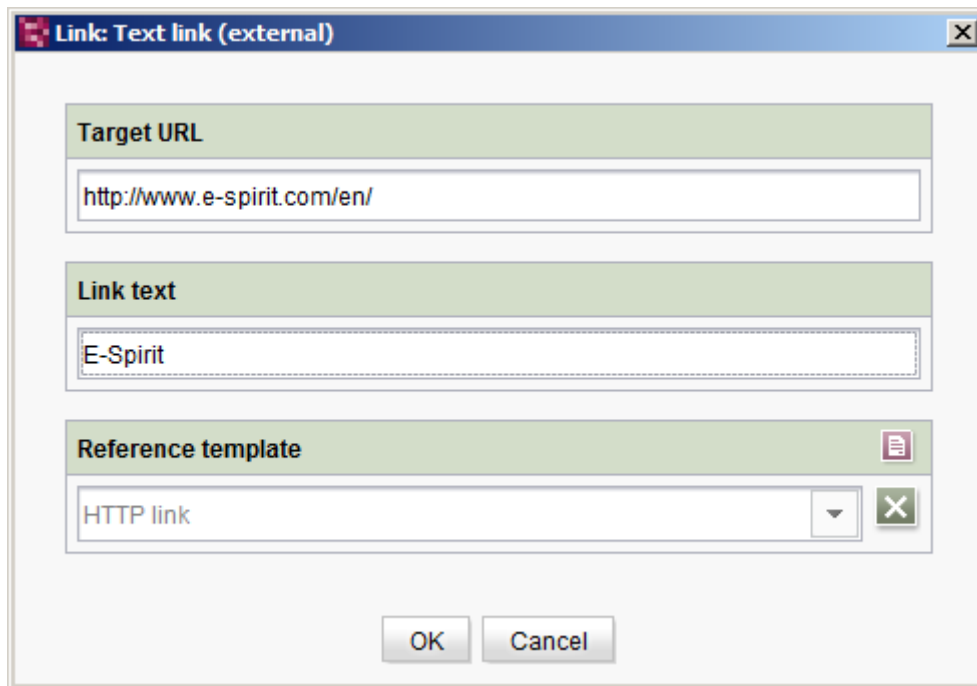
Figure 10-15: Select page reference

**Link text:** The link text to be displayed on the website can be entered in this field.





## 10.8.2 External link



The screenshot shows a dialog box titled "Link: Text link (external)". It contains three input fields:

- Target URL:** A text field containing the URL "http://www.e-spirit.com/en/".
- Link text:** A text field containing the text "E-Spirit".
- Reference template:** A dropdown menu with "HTTP link" selected and a list icon to its right.

At the bottom of the dialog are "OK" and "Cancel" buttons.

**Figure 10-16: External link**

This type of link is a link to a page of a published 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.e-spirit.com/de/>)

**Link text:** The link text to be displayed on the website can be entered in this field.

**Reference template:** Use this field to select which link template is to be used for creating the link.



### 10.8.3 Content Link



Figure 10-17: Database link



This type of link is a link to a dataset from the Content Store of the FirstSpirit project.

**Link text:** The link text to be displayed on the website can be entered in this field.

**Product link:** The select reference is now displayed in the top line with the object symbol and dataset ID. The content of the selected dataset reference is located below this. The icons for selecting and editing the selected dataset reference are the same as on the dataset selection input form (see section 10.10, page 294).

#### 10.8.4 Link to a related project

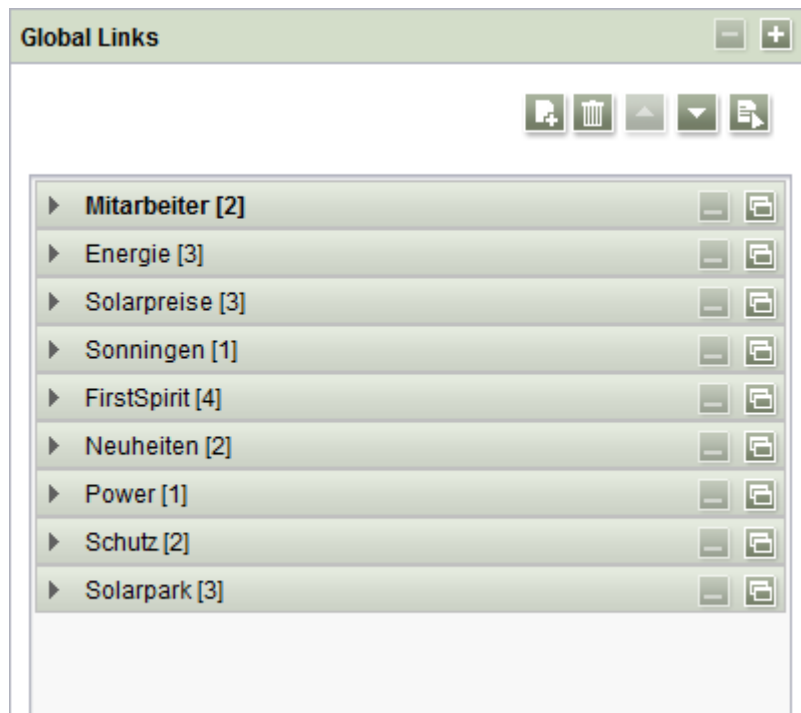
FirstSpirit supports remote access to other FirstSpirit projects. The remote access can be used within the current project to reference and display an element from the Site Store and/or Media Store of another FirstSpirit project (link to a related FirstSpirit project). The objects physically remain in the remote project.



*Remote access is a license-dependent additional module. A detailed description of this is given in the "FirstSpirit CorporateMedia" module documentation.*






## 10.9 List creation



**Figure 10-18: Input form – List creation**

This input form lets you select and display multiple sections as a list.

The content of all the list entries can be viewed and edited directly. 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 maximized or minimized using the icons  and  or  at the left border of the section overview or by double-clicking the window frame.

The icons for managing the list entries are located above the list and on each list entry.



**Add section:** click this icon to add a new element to the list. A selection dialog opens in which you can select the desired section template for the new list entry.



**Delete section:** click this icon to delete the selected section from the section list.





Move section: click this icon to move the selected section up/down in the list by one position.



Edit section: click this icon to show the selected section in the bottom area for editing.

In addition to normal sections and links, list creation can also be used to display and edit data from the Content Store.




ID	Picture	Picture name	Description
3072		Set top box	Set top box
2688		Inverters	Our inverters have a modular structure and can therefore be used in diverse ways.
960		Thin film modules	Thin film modules are particularly long-lasting.

Figure 10-19: Selection of datasets with list creation

The following icons are available, depending on the defaults set by the template developer and the application purpose of the input component:



Add (for datasets only): click this icon to open a selection dialog to select a dataset (see section 11.7, page 351).



New: in the case of datasets a window opens when clicking this icon to enter a new dataset, in the case of sections and links a selection dialog will open for selecting a correspondent template





Copy: in the case of datasets a copy of the selected dataset 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 (for datasets only): click this icon to delete the selected dataset from the list and from the related data source. This dataset 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 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 dataset 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:



If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)



## 10.10 Dataset selection

**Figure 10-20: Input form – Dataset selection**

Use this input form to reference a dataset from the Content Store. Selection is not limited to 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 form, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)



Select: click this icon to open a window for selecting a dataset from the data source.



Remove: click this icon to remove the selected dataset again.



Go to dataset: click this icon to switch to the Content Store for the selected source dataset.



Edit: click this icon to make changes to the referenced dataset.



Save: click this icon to save the changes made to the referenced dataset in the Content



Store.



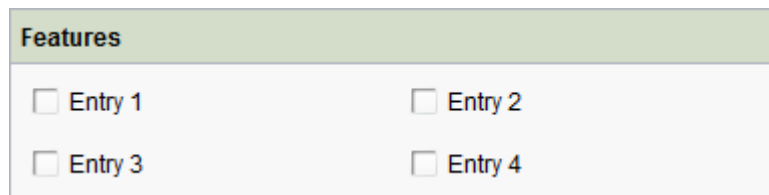
New: click this icon to add a new dataset to the data source.



Show language tab: click this icon to display the tabs for all project languages for the dataset; click it again to hide the tabs.

The **Show usages** button can be used to determine at which position the current dataset is referenced.

## 10.11 Checkbox



Features	
<input type="checkbox"/> Entry 1	<input type="checkbox"/> Entry 2
<input type="checkbox"/> Entry 3	<input type="checkbox"/> Entry 4



**Figure 10-21: Input form – Checkbox**

This input form does not allow users to enter content directly; instead, users can select from values specified by the project developer.

Checkboxes let you

- make **more (or less) than one selection** at the same time
- leave the selection **blank**.

When one checkbox is selected, any checkboxes selected previously are **not** disabled.

If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)





## 10.12 Selection list

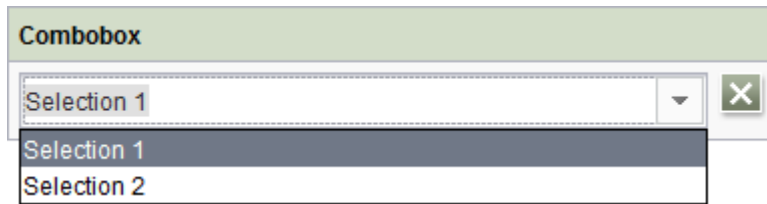





Figure 10-22: Input form – Selection list

Using a selection list (combobox), the user can

- select **exactly one** value from a group of values specified by the project developer, or
- leave the selection **empty**.

Depending on the configuration, the user can enter a **custom value** as well.

The entry can be selected by clicking on the arrow symbol next to the entry line. The desired value can be selected from the selection list when it opens. The selection can be reset using the  icon.

If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)

## 10.13 Radio button

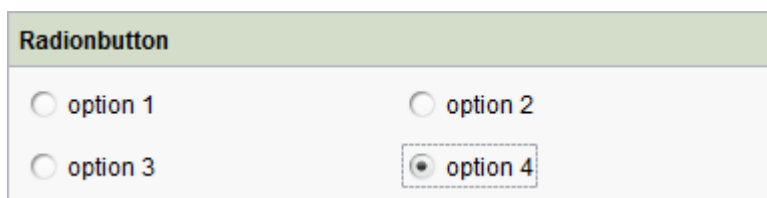


Figure 10-23: Input form – Radio button



This input form does not allow users to enter content directly; instead, users can select from values specified by the project developer.

A radio button allows users to

- choose **precisely one** value from a set of values.

When one field is selected, any checkbox selected previously is disabled.



If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)

## 10.14 Toggle



**Figure 10-24: Input form – Toggle**



This input form does not allow users to enter content directly; instead, users can select from values specified by the project developer.

A toggle allows users to

- choose between **precisely two** specified values (e.g. on/off, right/left).

If one field is activated, the other is disabled.

The toggle can be shown as a radio button or a checkbox.

If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)



## 10.15 Multiple selection list





**Figure 10-25: Input form – Multiple selection list**

The user cannot enter content directly using this input form, but it is possible to select values that are specified by the project developer.

Using a multiple selection list, the user can

- select **more (or fewer) values** from a group of values at the same time, or
- leave the selection **empty**.

The entry can be selected by clicking on the arrow symbol next to the entry line. The desired value(s) can be selected from the selection list when it opens. Selected values can be removed again by using the context menu (right-click with the mouse).

If the project developer has defined a preset default value for the input component, the "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)



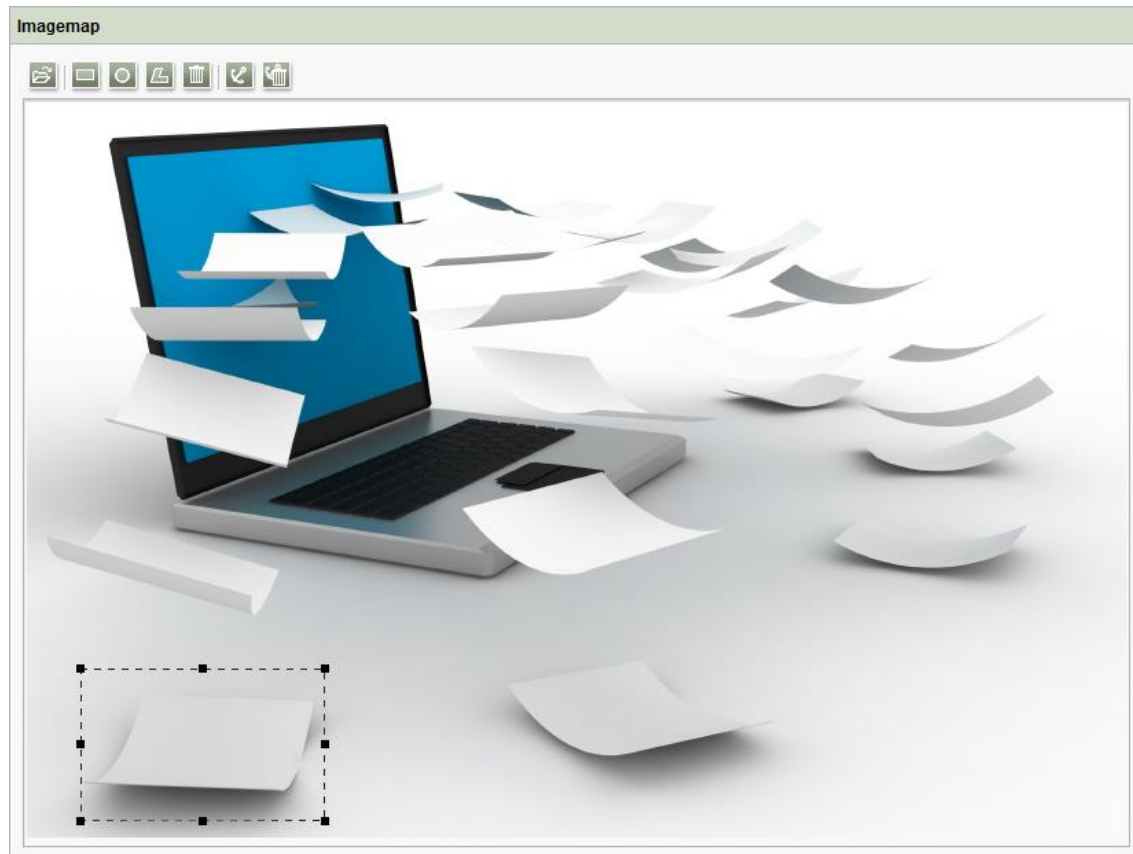
## 10.16 Button

This input form is shown as an icon, button or link, depending on the template developer's specification. Clicking it performs a function defined by the template developer. Depending on the configuration of the input form, 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.

Depending on whether the input form is displayed as a button, icon or link, it can have one of the three following appearances:





## 10.17 Link-sensitive graphic



**Figure 10-26: Input form – Link-sensitive graphic**

This input component can be used to integrate references (links) to various positions of a selected background image. 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 "Reset to default value" icon  or the "Adopt default value" icon  is located in the top right-hand corner. (For further information, see section 10.18, page 302.)



Select background image: click this icon to open a selection dialog for selecting the background image from the Media Store.



Additionally, you have to specify in the **Resolution** field which of the available resolutions to use for the background image.



Generate rectangle; click this icon to generate a frame object in the form of a rectangle on the background image. 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 image. 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 image 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; if 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 form (see section 10.8, page 285). 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.



## 10.18 Default values and invalid entries

### 10.18.1 Default values

The "Reset to default value"  icon can be seen in the top right-hand corner of input forms. This icon always appears if the template developer has specified a default value for this input form.

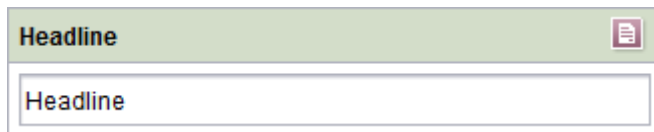

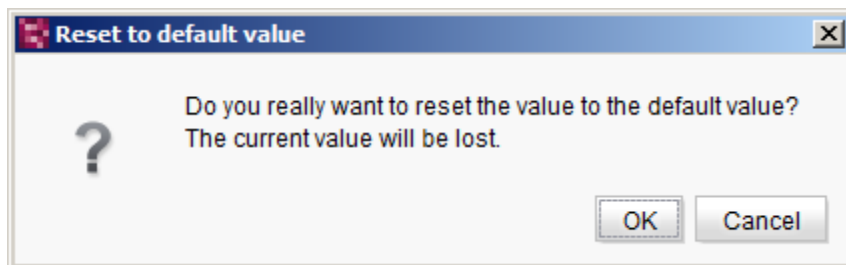


Figure 10-27: Input component with the "Adopt default values" icon

Click the  icon to reset the content of the input component back to the default 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 default value.


The input component is surrounded with a different-colored frame to indicate that its content is a default value. The "Adopt default value"  icon now appears in the top right-hand corner



Figure 10-28: Input component with default value



## 10.18.2 Invalid entries

To help editors create editorial content, the template developer has the option of specifying rules for entering content in an input form. If a rule is violated when content is entered, the appearance of the input form changes to highlight the fact that a faulty entry has been made.

The faulty entries are highlighted in color and text with a reason for the fault appears.

The color highlighting depends on the available restriction level of an invalid entry.

- Maximum restriction level: Not possible to save an invalid entry – highlighted in red
- Medium restriction level: Possible to save but not release an invalid entry – highlighted in yellow
- Minimum restriction level: Save and release possible, for information only – *not* highlighted

In addition to the color highlighting of the input form, the template developer can also define a correction note that appears directly below the input component in question.



**Figure 10-29: Color schema and correction note**

An additional correction note with the request "Please correct your entry" is displayed within the workspace for the entire section. It always remains visible to the editor.



**Figure 10-30: Correction notes for the section**

The number of entries that need correcting (within a section) is displayed to the editor in the "Display [number]" button. Clicking this button opens a list of all the input forms in the current workspace that still contain invalid entries. Clicking on the desired entry takes the editor straight





to the input component in question in order to correct the entry.

## 10.19 Enhanced functions in input forms

To make working with the input forms in FirstSpirit more convenient, some functions have been added. These can be reached via a context menu.

To make transferring content easy, all input forms now have a context menu that can be used to cut, copy and paste the content of the input form. Right-click the title of the frame of the input form to access this context menu.

Alternatively, use the following keyboard shortcuts:

- <Ctrl> + X (Cut)
- <Ctrl> + C (Copy)
- <Ctrl> + V (Paste)

If reasons of incompatibility make it impossible to copy values from one input form to another, the "Paste" (Ctrl+V) entry is deactivated.

In text-based input forms, such as single-line text, multi-line text, numbers, date and link input, the following functions can also be accessed by right-clicking the text field:

- **Search:** Opens a dialog for finding a specific text in the input forms.
- **Replace:** Opens a dialog for finding a specific text in the input forms and replacing that text.
- **Next occurrence:** If you searched for a specific text, this entry jumps to the next hit in the input form. The search does not stop at the end of the input form.
- **Find previous:** If you searched for a specific text, this entry jumps to the previous hit in the input form. The search does not stop at the beginning of the input form.
- **Undo:** Resets all changes in the input form since Editing mode was activated.
- **Redo:** Restores any changes reset with Undo.
- **Paste:** Pastes content from the clipboard at the cursor's position.
- **Copy:** Copies selected content into the clipboard.
- **Cut:** Removes the highlighted content and copies it into the clipboard



## 11 General use

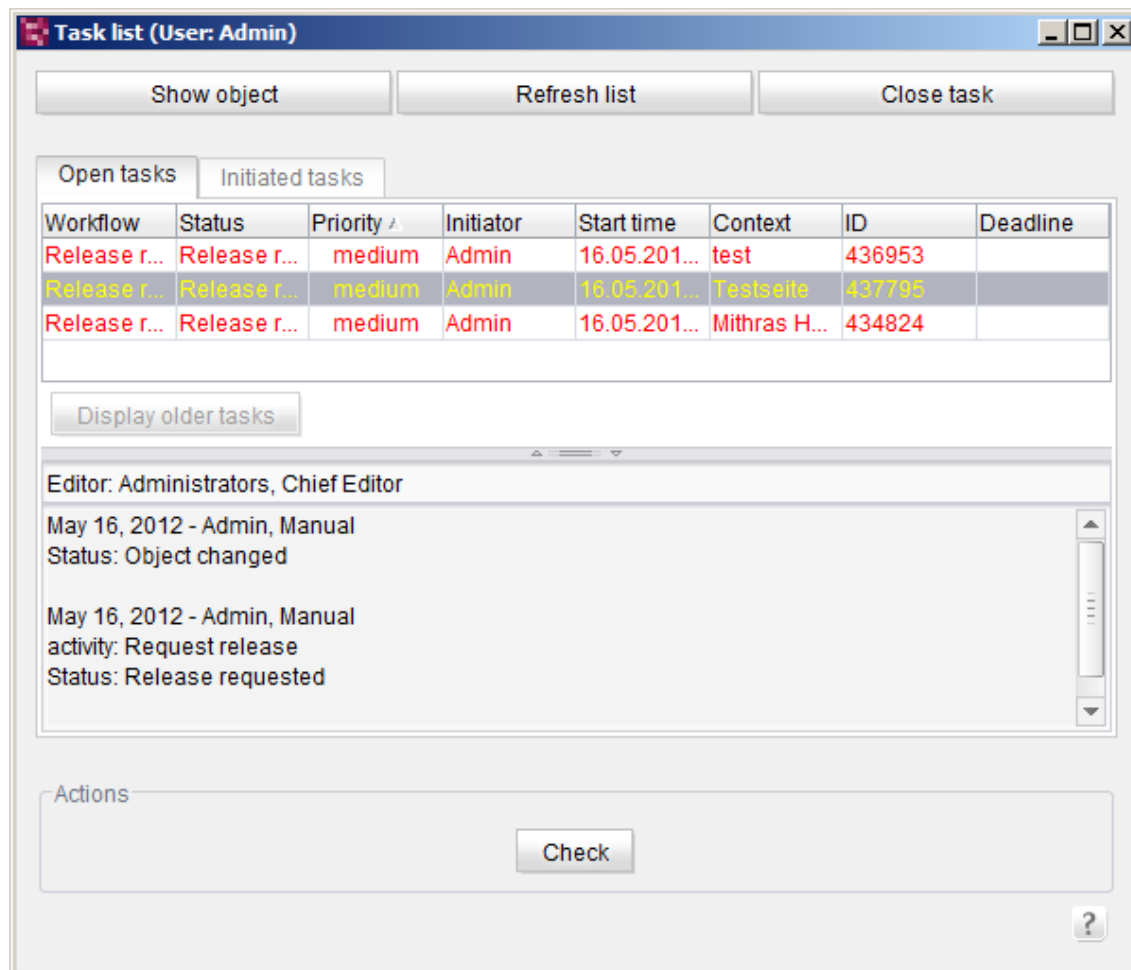
### 11.1 Task list

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" tool bar or CTRL + T).

The **Open tasks** tab contains all the tasks to be dealt with by the user who is currently logged on. 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 user who is currently logged on. 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**

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 color for improved clarity. Black text means that the user is not directly selected as the editor for this task. Red lettering means that the user is either directly selected as the editor or belongs to a group that is the selected editor.

Invalid tasks, which result e.g. because an object on which a workflow is active is deleted, are visualized 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 color 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 synchronize the task list of the FirstSpirit Server with the open task list.



*If you attempt to carry out an action on a task that is not up-to-date, the following message appears: "The element was outdated and has been updated." "Do you still want to continue?" If you answer "Yes", the updated action window opens if further action is possible. If no further action on the updated task is possible, another error message appears: "This object was changed by another user" Please try again."*

Click the **Close task** button to close tasks without running through the associated workflow. This could be necessary if a task has become invalid (e.g. because the object on which the workflow is active has been deleted).

Open tasks are also displayed on the project entry page (see section 3.6, page 124).

## 11.2 The message board

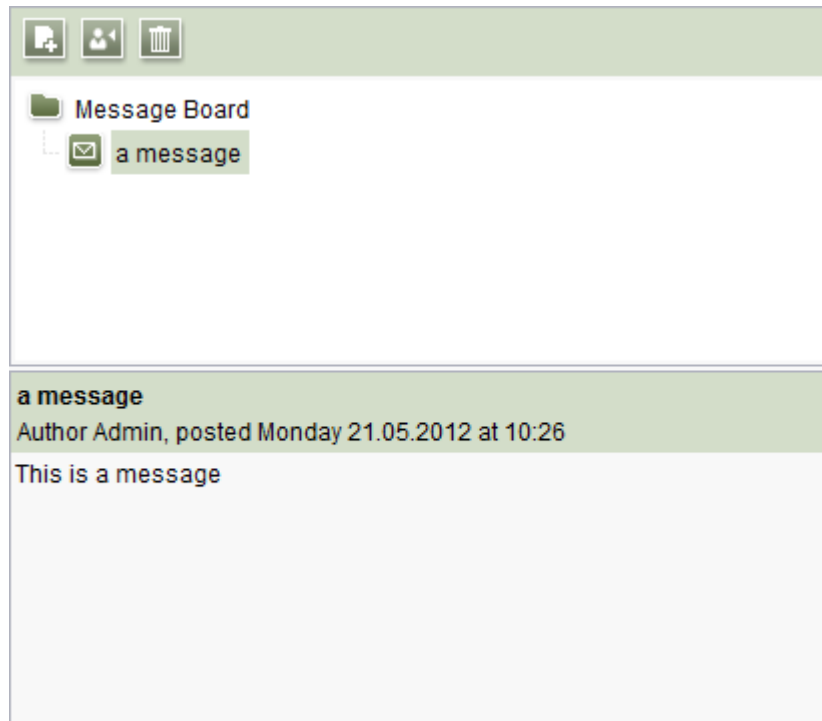
There are message boards in several places in the client. The message board function is identical everywhere. The message boards are located in the following places in the individual Stores:

- 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. For example, 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.



**Figure 11-2: Message board**



Click this icon to create a new message. A window opens in which you can enter a message subject and a message text.



Click this icon to reply to an existing message. 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 this icon to delete the currently selected message and all the linked replies, without a confirmation prompt.

Click on a message within the tree structure to access the content of the message. 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.



## 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 keyboard shortcut "**ALT + P**". For example, the number of integrated media can be ascertained in this way.

Apart from this 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 section 13.1.4.11, page 425). Changes can only be made to the metadata if it is 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 section 11.4.3, page 311), the respective source section or the object chain of the source section will be used to determine the meta information for evaluation methods 1 and 2, and the object chain of the section reference for evaluation method 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 entered metadata is indicated by a " ⓘ " after the object name.



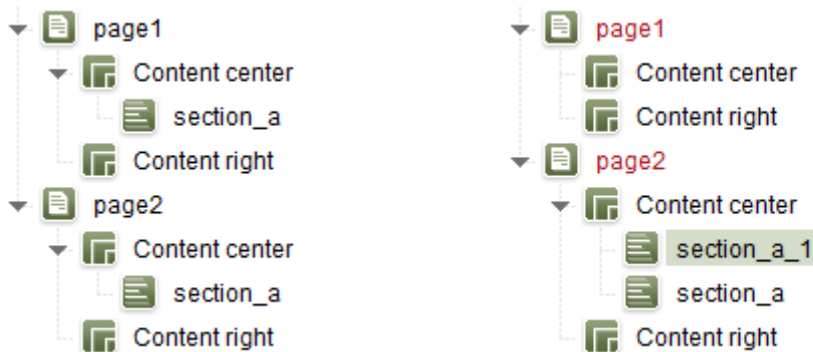
## 11.4 Drag-and-drop functions of FirstSpirit

FirstSpirit offers a variety of drag-and-drop functions to make working with it more user-friendly.

### 11.4.1 Move using "drag-and-drop"

The objects in the stores (folders, pages, media, etc.) can be moved by drag-and-drop with the mouse.

When moving sections of the Page Store, 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 11-3: 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 section 3.1.6.1, page 65). 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.*

### 11.4.2 Copy using "drag-and-drop"

If you hold down the Ctrl key, the objects in the stores (folders, pages, media, etc.) can be copied by drag-and-drop with the mouse (indicated by a small plus sign 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 11-4: Example – Copying a section with the same reference name

### 11.4.3 Create section references in the Page Store

Section references can be used to reuse the content of a section on other pages. They are created by moving the section to be referenced ("source section") using your mouse (drag-and-drop) by simultaneously pressing the SHIFT and Ctrl keys (denoted by a small arrow on the mouse pointer). A reference to the section is created in this way.

### 11.4.4 Drag-and-drop from the local file system into JavaClient

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-and-drop.

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.

- Image selection (CMS\_INPUT\_PICTURE)
- File selection (CMS\_INPUT\_FILE)
- Reference selection (FS\_REFERENCE)

A window for selecting an upload folder opens directly in the Media Store. After you confirm your selection, the medium is inserted in both the selected Media Store folder and in the desired input component.



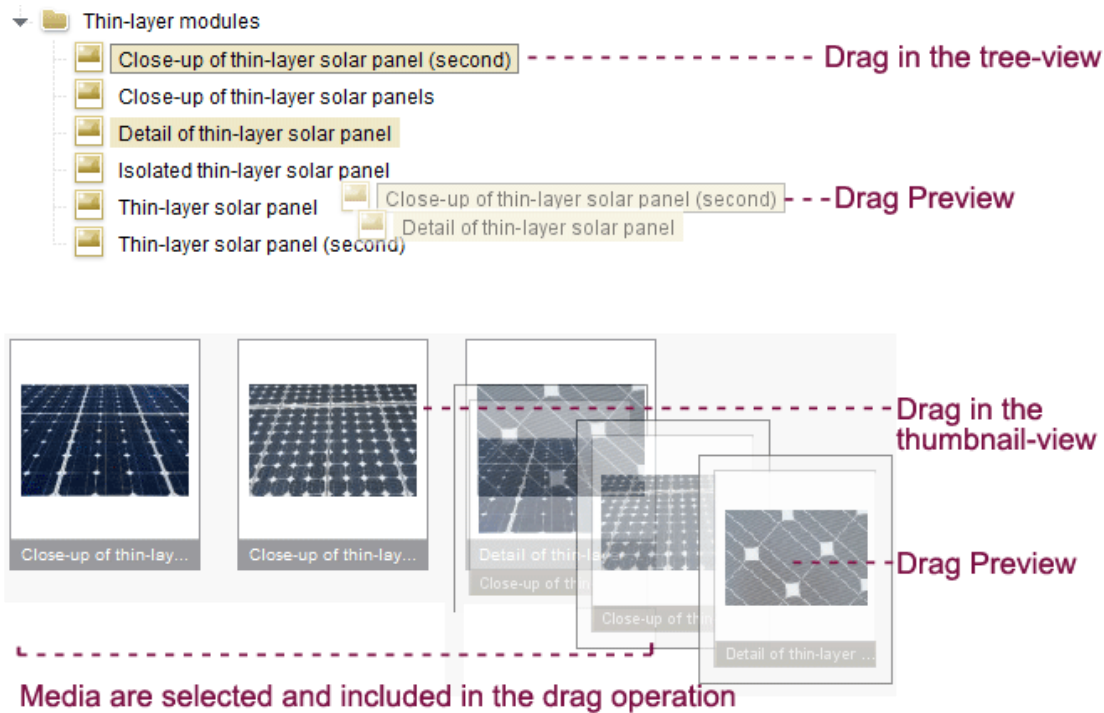


### 11.4.5 Drag-and-drop from JavaClient into the local file system

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-and-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-and-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 11-5).



**Figure 11-5: 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 11-6):



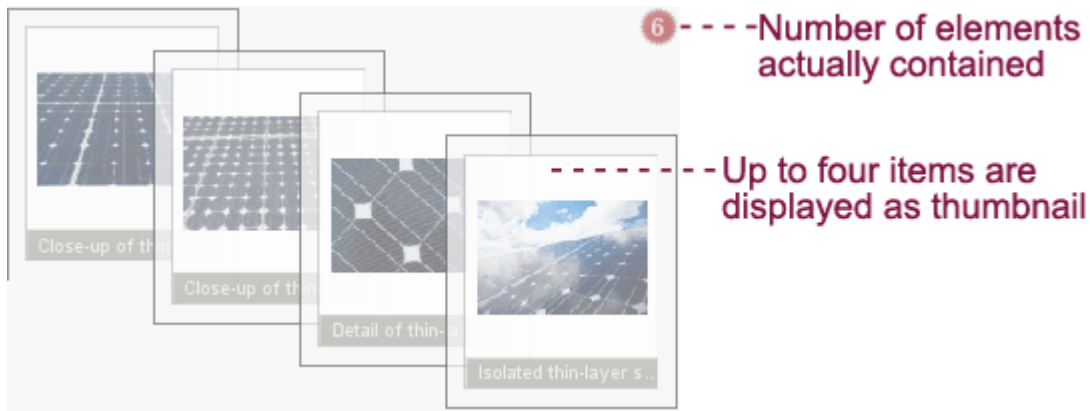


Figure 11-6: Drag action with more than four elements

### 11.4.6 Drag-and-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-and-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).

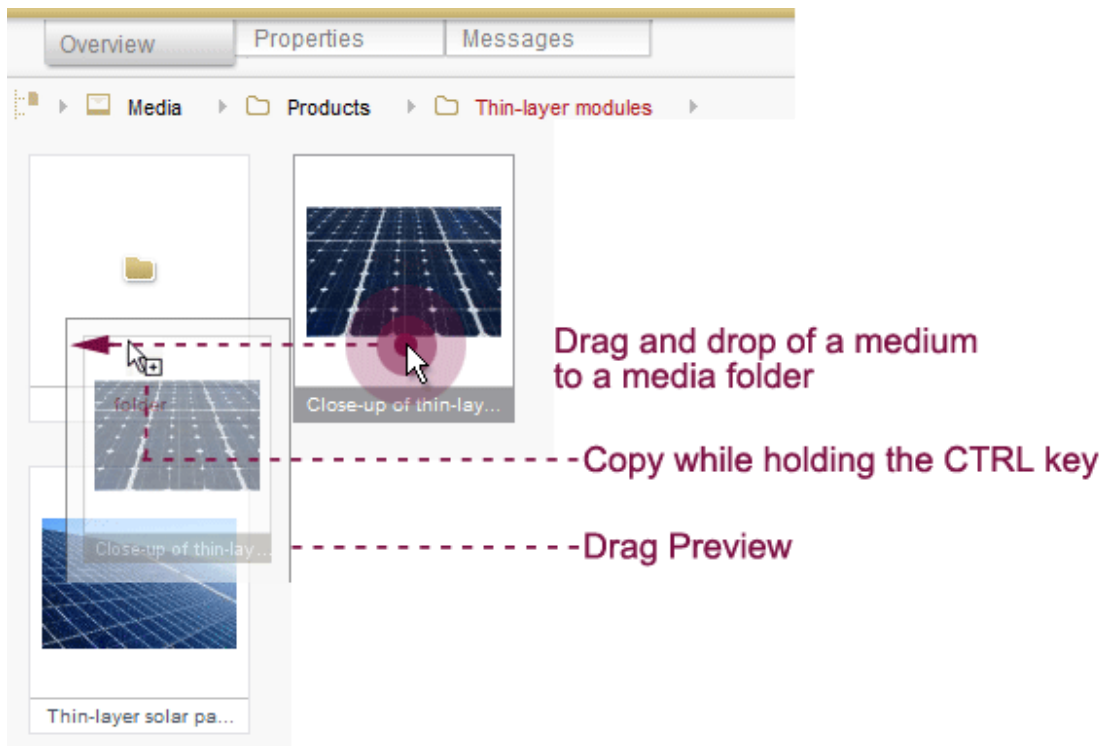


Figure 11-7: Drag-and-drop media in the thumbnail view




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-and-drop" action to copy media into a folder. The copying of media (or folders) is visualized by a + symbol on the mouse cursor. Simple drag-and-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.

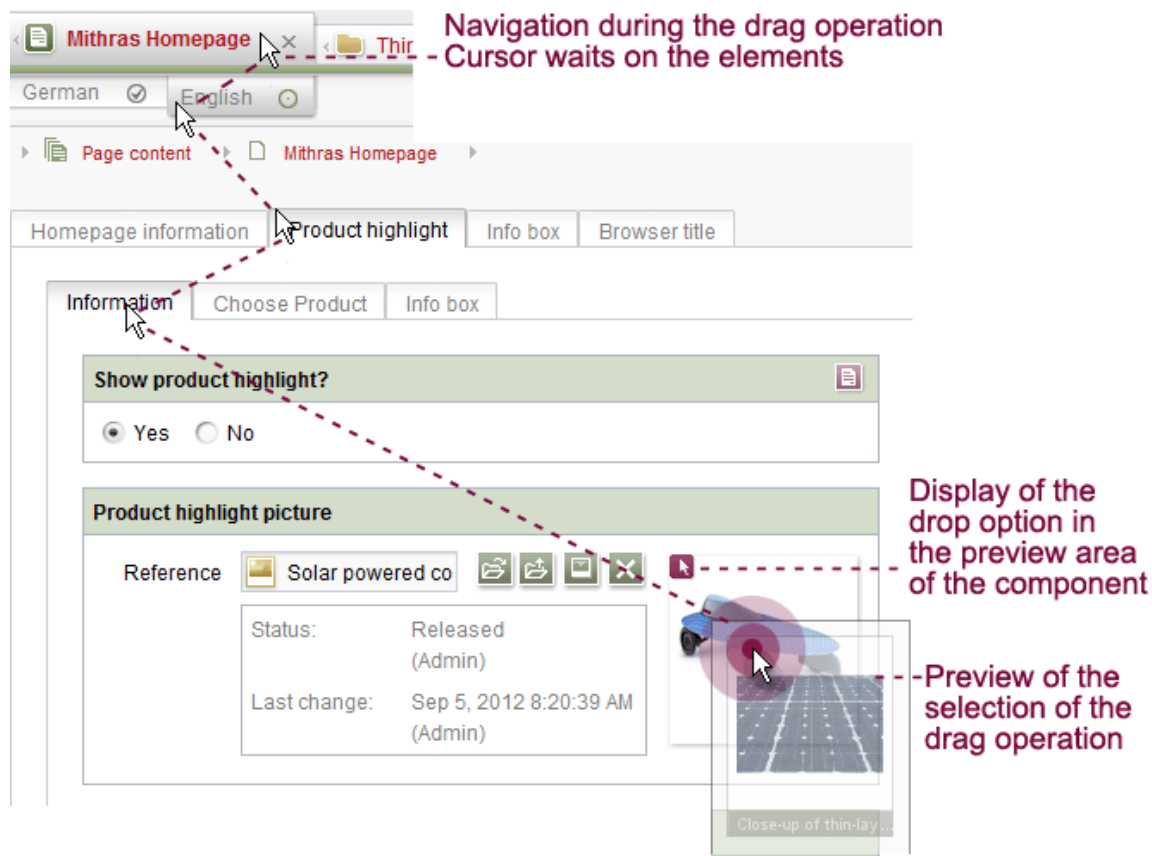
#### 11.4.7 Drag-and-drop between two workspaces

Apart from editing in individual workspaces, it is also possible to drag-and-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-and-drop (see section 3.5.1, page 120).


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 drop 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, navigation can also be performed on the tabs of grouping elements of a form (see Figure 11-8).





**Figure 11-8: 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.

The individual workspaces can also be moved within the horizontal tab navigation by means of drag-and-drop.

A new workspace is opened for elements that were moved to the horizontal tab navigation from the tree structure by drag-and-drop.

### 11.4.8 Inter-Store drag-and-drop

It is also possible to drag-and-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-and-drop between two workspaces" (see section 11.4.7, page 314).

The inter-store drag-and-drop function can be used to easily integrate folders and pages from the Page Store in the Site Store as a new menu level or page reference.

#### 11.4.9 Drag-and-drop from the search dialog into JavaClient

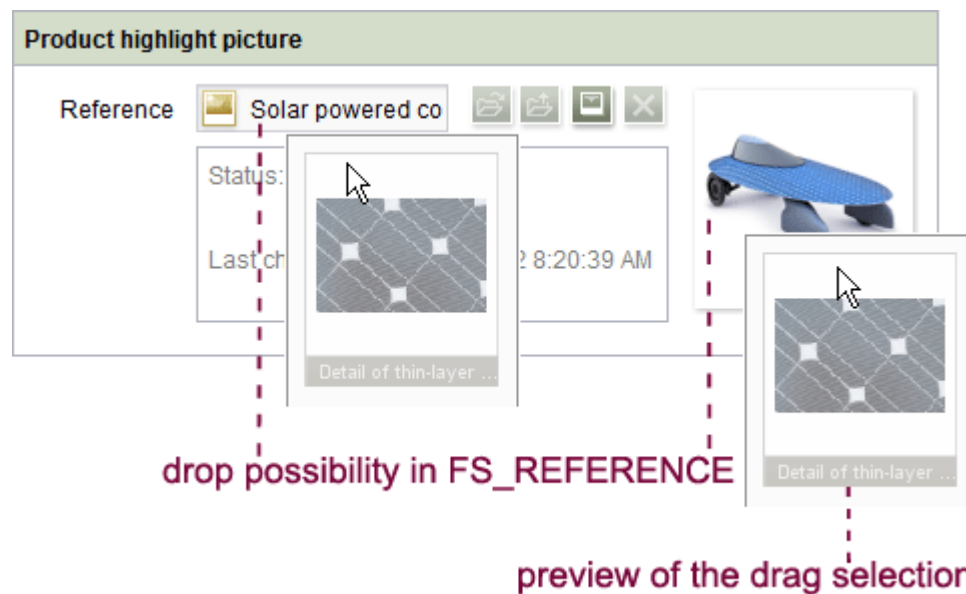
The search dialog for the quick text search (see section 3.3.1, page 90) also supports drag-and-drop of the search results into JavaClient. The FirstSpirit objects found can be dragged directly from the search dialog into the tree view of JavaClient, the thumbnail view for media, the workspace or an input component. In this way, for example, the editor can drag a medium directly from the search dialog into the image input component of a workspace and drop it there (see Figure 3-55).

#### 11.4.10 Drop onto media input forms

Drag-and-drop is also supported for selecting media in the reference selection input form:

The procedure on switching workspaces has already been described under "Drag-and-drop between two workspaces" (see section 11.4.7, page 314). The object can then be dropped into the preview area or the reference field of the input form.





**Figure 11-9: Drop into an image input form**

If the template developed defined restrictions to certain folders in the Media Store for the input form, only media from these folders may be referenced in the input form. 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 form.

#### 11.4.11 Drop onto text input forms

Drag-and-drop is also supported for entering text in some input forms:

- Rich text editor
- Single-line text
- Multi-line text

Selected text sections can be inserted into an input form by drag-and-drop. These text sections can come from other input forms within the FirstSpirit project or from other FirstSpirit projects. Additionally, text sections from external files such as Word files or PDF documents and websites are also possible.

#### 11.4.12 Drag-and-drop within input forms

The drag-and-drop function is also supported within some input forms.

- Rich text editor
- Component grouping



Within the **rich text editor**, the selected text (including all formatting) can be easily moved by holding down the left-hand mouse button (indicated by a small rectangle on the mouse pointer). If you hold down the Ctrl key at the same time, the selected text is inserted in the desired position as a copy (indicated by a small plus sign on the mouse pointer).

Within the **component grouping**, you can also use drag-and-drop across two tab pages. To do so, you select a medium or text on one tab page and then hold down the left-hand mouse button and drag it to the tab of the desired tab page. If you hover the mouse pointer over the tab, the tab page opens and you can insert the medium or text in a suitable input form. Note that in this case the content is copied (indicated by a plus sign on the mouse pointer). To move the content, hold down the Ctrl key at the same time (indicated by a rectangle on the mouse pointer).

#### 11.4.13 Drag-and-drop between two input forms

Drag-and-drop operations are also possible between two input forms.

In the case of **input forms with a reference** to other objects in the project, you can drag-and-drop a reference symbol to copy the referenced object to the second input form.

To do this, select the preview or type icon within an input form and hold down the left-hand mouse button to drag it to the reference field or preview area of the second input form.

In the case of **text input forms**, you can use drag-and-drop to copy the selected text to the second input form.

- Rich text editor
- Single-line text
- Multi-line text

The selected text can be easily moved by holding down the left-hand mouse button (indicated by a small plus sign on the mouse pointer). If you hold down the Ctrl key at the same time, the selected text is moved to the second input form (indicated by a small rectangle on the mouse pointer).

The selected text and formatting is copied in the rich text editor. The system checks whether the format to be copied is allowed in this component; formats that are not allowed are copied unformatted as plain text.

In the case of selected text that does not come from another text input form, the system attempts to generate and insert a "toText" representation. (Examples: Selection list → selection text in the



current language; reference selection → display name)

### 11.4.14 Drag-and-drop whole input forms

It is now also possible to move input forms in their entirety by drag-and-drop. The values from input forms are also copied to the input forms. The best way to do this is to click on the highlighted title of an input form and hold down the mouse button while you drag it to the desired target input form. In the case of input forms that do not have a highlighted title, you can click on the title, for example. A purple insertion mark indicates locations where the input form may be dropped.

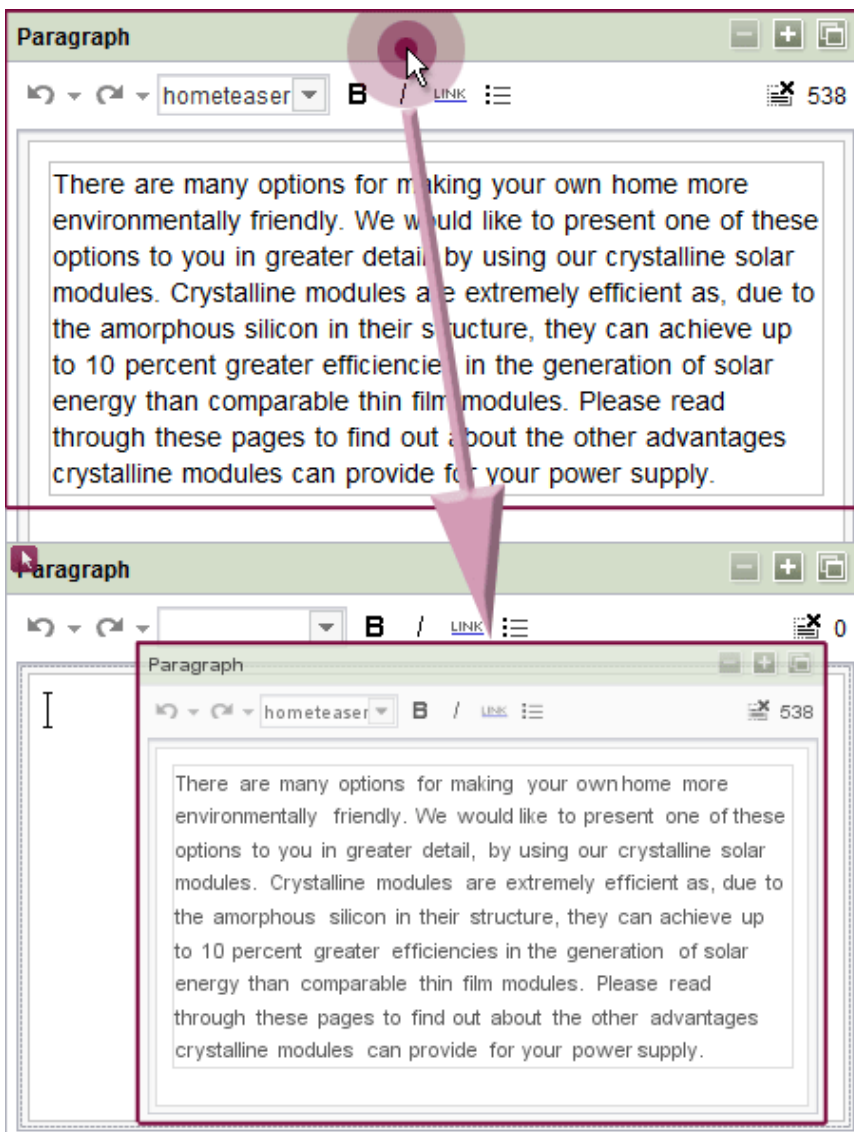


Figure 11-10: Drag-and-drop input components





In principle, this is possible for input forms of the same type. If the destination is an input form that can accept text (e.g. rich text editor, single-line text, multi-line text), the system attempts to convert the information from the element to be dropped into text and then insert it, e.g. the name of a medium, text for a selection option, etc. If the destination is also an input form (drop on the tile), any existing value is overwritten; if the destination is a text field (for example), the value is extended. In the case of a rich text editor, formatting is also copied if the configuration permits. Other input forms, such as Numbers or Date can be copied and pasted together with their content by drag-and-drop.

#### 11.4.15 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. As with navigation between workspaces during a "drag" action (see Figure Figure 11-8), the navigation is also carried out via the tree view.

If the mouse cursor lingers

- over the object symbol, the node in the tree is expanded
- over the labeling 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-and-drop between two workspaces (see section 11.4.7, page 314).

#### 11.4.16 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 reference selection, a message appears 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 form.

#### 11.4.17 Drag-and-drop via the Content Store overview

When using drag-and-drop on FirstSpirit objects in the Content Store, the required editing area can be opened via the overview of the datasets (see also section 5.5, page 163). 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.

#### 11.4.18 Drag-and-drop from the integrated preview

You can also drag media using drag-and-drop from a page of the integrated preview into the workspace of the Media Store. In this case, the function "Content highlighting control" should be deactivated in the menu "View".

If you drop the image onto one of the resolutions of an existing image, the dropped image will be directly inserted for the selected resolution. If you drop the image onto a folder of the Media Store, you will first be asked for the display and the reference name of the new image, then the new image will be inserted into the selected folder of the Media Store.

#### 11.4.19 Drag-and-drop from Microsoft applications

A block selection can be copied to the JavaClient by drag-and-drop from most Microsoft applications (Word/IE and FF). Depending on the configuration and the destination of the drop action, different rules apply for copying formatting information:

- Word import not configured and drop into rich text editor: A firmly implemented default rule set is used to map HTML fragments from FirstSpirit structures as far as possible (e.g. bold, italic, lists, table structures). Section/character formats, images and links, for example, are not mapped.
- Word import configured and drop into rich text editor: The system asks you which import rule to use. If the configuration is ideal, complex formatting can be copied (e.g. section/character formats, images, links and tables, including simple formatting).
- Drop into simple text input components: A "toText" representation is inserted.



## 11.5 Working with the rich text editor input form

The rich text 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 icon bar, the context menus and the corresponding keyboard shortcuts for the rich text editor are explained in this manual.

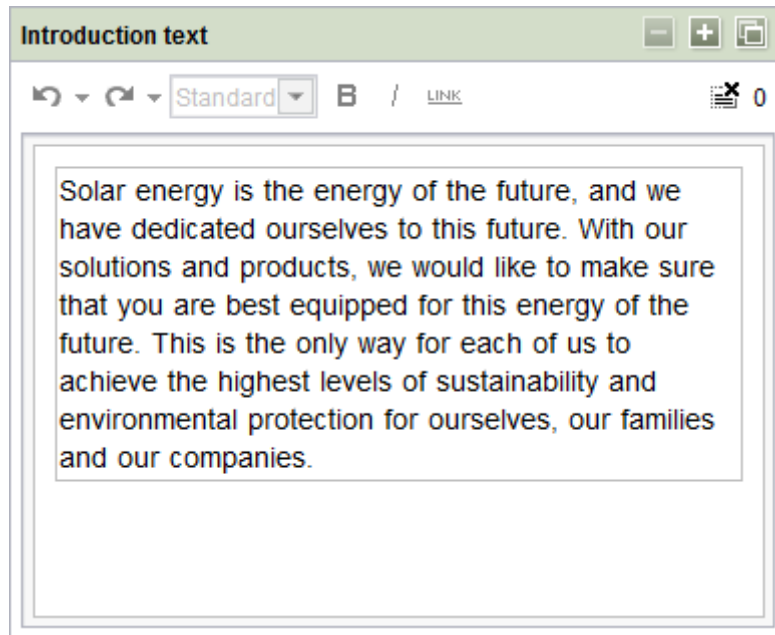


Figure 11-11: DOM Editor input component

### 11.5.1 Icon bar in the rich text editor

The standard functions which can be opened via the icon bar have already been described in the rich text editor section under Standard Input Components. See section 10.3, page 275.

### 11.5.2 Context menu in the rich text editor

#### 11.5.2.1 Context menu on selected 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.

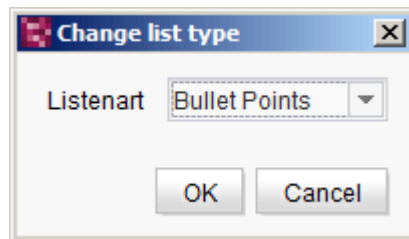


**Cut:** This function is used to cut the selected text of the section and copy it into the clipboard.

#### 11.5.2.2 Context menu within a list

**Move item one level up:** This function can be used to move the highlighted list item up by one level.

**Change list type:** This function can be used to change the list type.



**Figure 11-12: 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 image of the selected list type is to be displayed on the website.

**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.

#### 11.5.2.3 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.

#### 11.5.2.4 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 section 10.8, page 285.

**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.

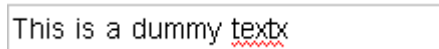


### 11.5.3 Spell check in the rich text editor

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.

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 11-13: Display unknown words in the input component**

The user can open the context menu by right-clicking the word highlighted in red:

**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 occurrences of the word in the input component. The word is no longer highlighted in red, but it will not be added to the dictionary. If the spell check is started again the word is 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.

## 11.5.4 Lists: Generating lists

### 11.5.4.1 Generate a new empty list

Click the **Insert list**  icon to generate a new, empty list.



**Figure 11-14: Generate new, empty list**

The cursor is then in the new empty bullet point.

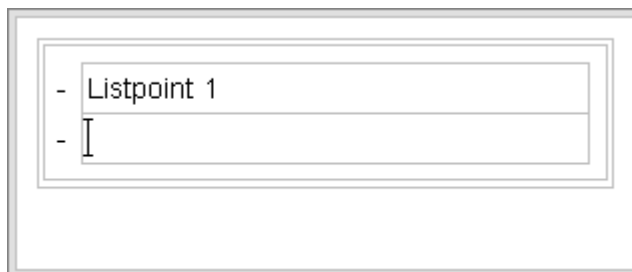


## 11.5.4.2 Add new bullet point



Press <RETURN> at the end of a bullet point to add a new empty bullet point.

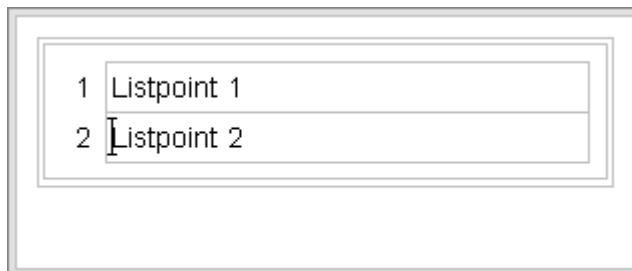
**Figure 11-15: Add bullet point 1**



**Figure 11-16: Add bullet point 2**

The cursor is then in this empty bullet point.

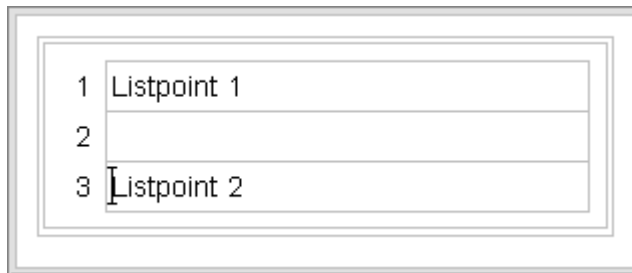
## 11.5.4.3 Add several new bullet points



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 11-17: Several bullet points 1**



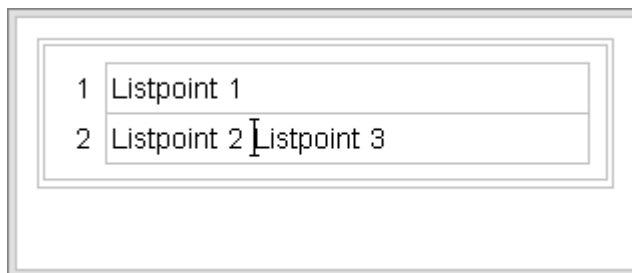


Any number of empty bullet points can be generated in this way.

**Figure 11-18: Several bullet points 2**

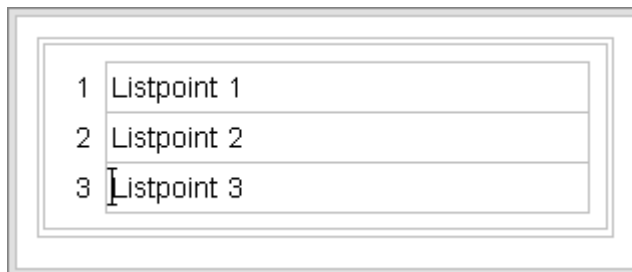
The cursor is then still at the start of the bullet point which is not empty.

#### 11.5.4.4 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 11-19: Add bullet point with content 1**



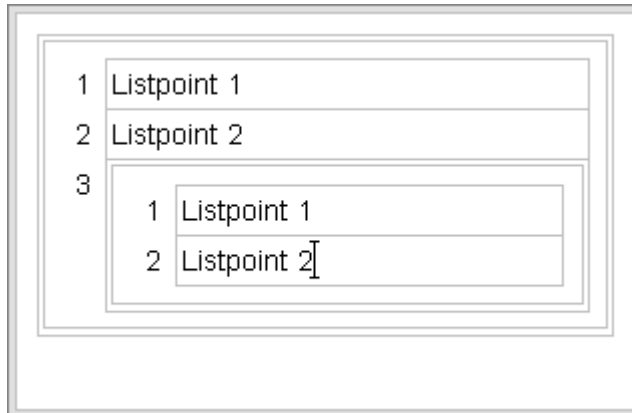
**Figure 11-20: Add bullet point with content 2**

The cursor is then at the start of this new bullet point in front of its content.



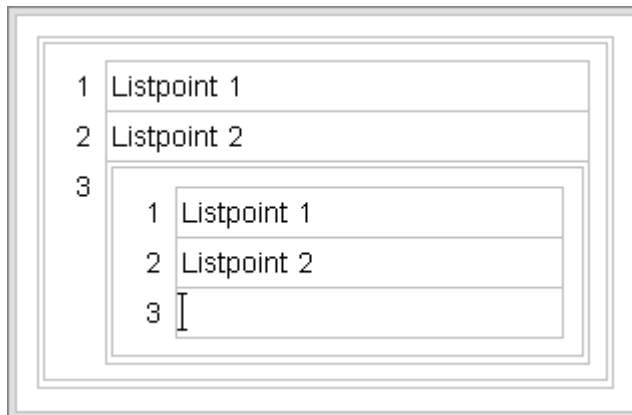


## 11.5.4.5 Add new bullet point to a nested list



Press <RETURN> at the end of a nested bullet point to add a new empty bullet point with the same nesting depth.

**Figure 11-21: Add nested bullet point 1**

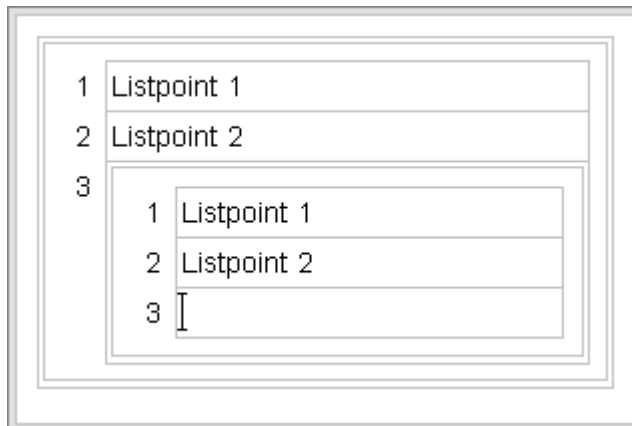


**Figure 11-22: Add nested bullet point 2**

The cursor is then in the new empty bullet point.

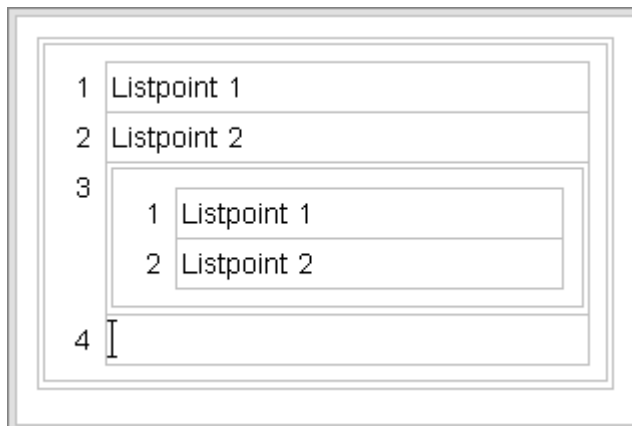


## 11.5.4.6 Add new bullet point behind a nested list



Press <RETURN> in an empty bullet point of the 2nd list level to move the empty nested bullet point to one level higher.

**Figure 11-23: Increase bullet point level 1**



**Figure 11-24: Increase bullet point level 2**

The cursor is then still in the empty bullet point.



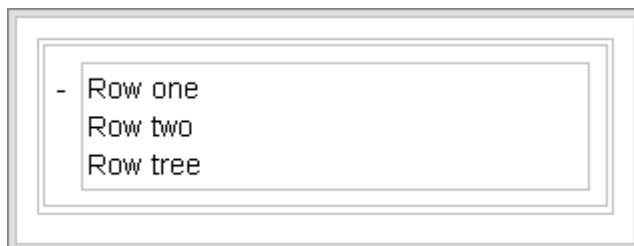
## 11.5.5 Convert selected text into list

### 11.5.5.1 Complete highlighting 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 11-25: Convert complete section into list 1**



**Figure 11-26: Convert complete section into list 2**

The list consists of one bullet point.

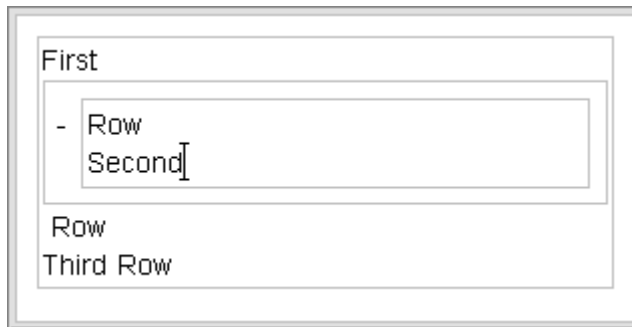
### 11.5.5.2 Partial highlighting within a section



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 11-27: Partially convert section into list 1**

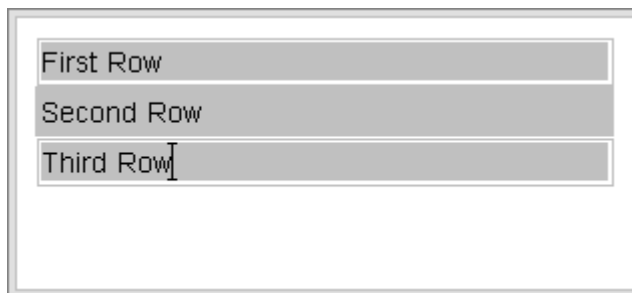




**Figure 11-28: Partially convert section into list 2**

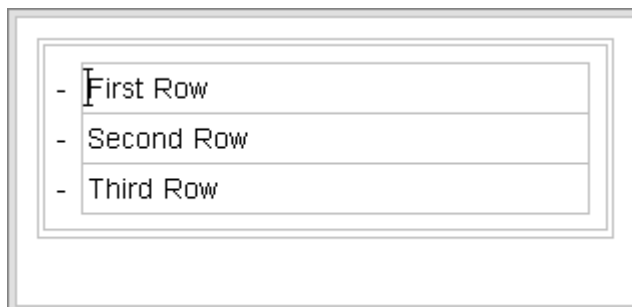
The list consists of one bullet point. The text fragments which are not highlighted are kept in the same section in front of or behind the list.

### 11.5.5.3 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 11-29: Convert several sections into list 1**

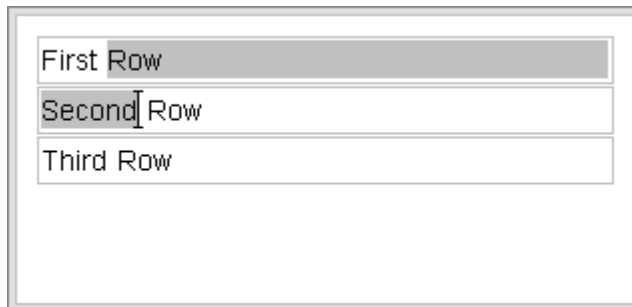


**Figure 11-30: Convert several sections into list 2**

If there are three existing sections the list then also consists of three bullet points.

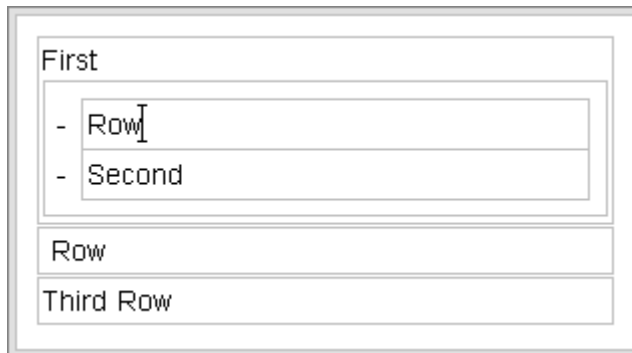


## 11.5.5.4 Partial highlighting of several sections



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 11-31: Partially convert several sections into list 1**



**Figure 11-32: 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.



## 11.5.6 Exit/interrupt list

### 11.5.6.1 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 11-33: Exit list at the end 1**

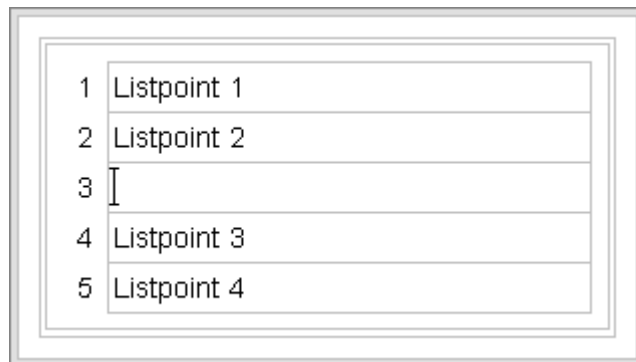


**Figure 11-34: Exit list at the end 2**

The cursor is then in this empty standard section.

### 11.5.6.2 Divide list between two bullet points

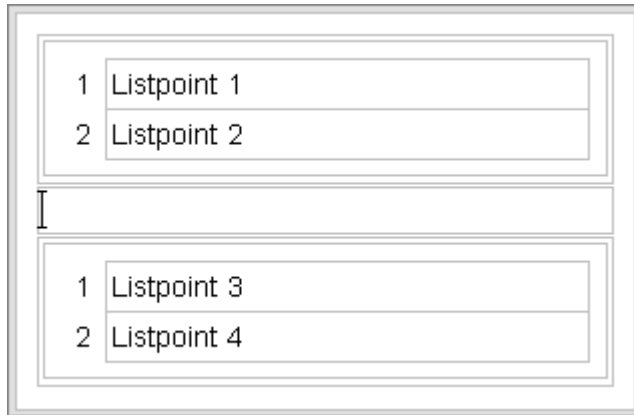
To divide a list an empty bullet point must be inserted at the required position.



Then press <RETURN> to divide the list at this point and the numbering of the existing bullet points is reset accordingly.

**Figure 11-35: Divide list 1**



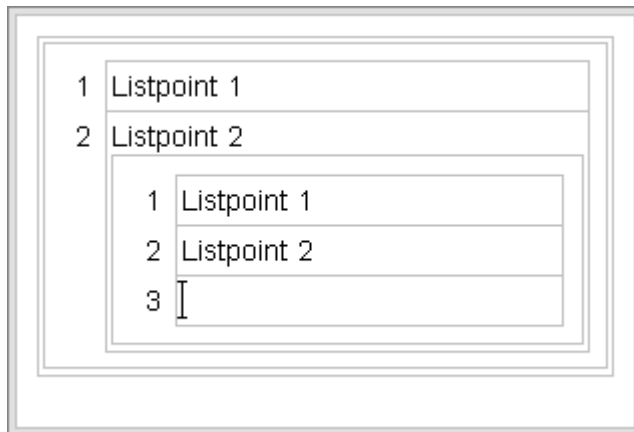


An empty standard section is inserted between the two lists.

**Figure 11-36: Divide list 1**

The cursor is then in this empty standard section.

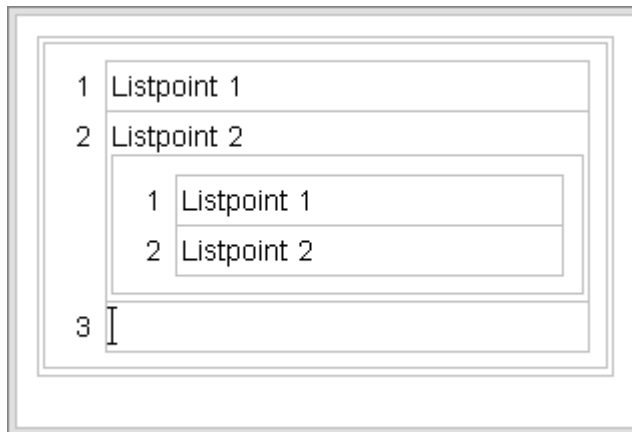
### 11.5.6.3 Divide list at the end of a nested list



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.

**Figure 11-37: Divide nested list 1**



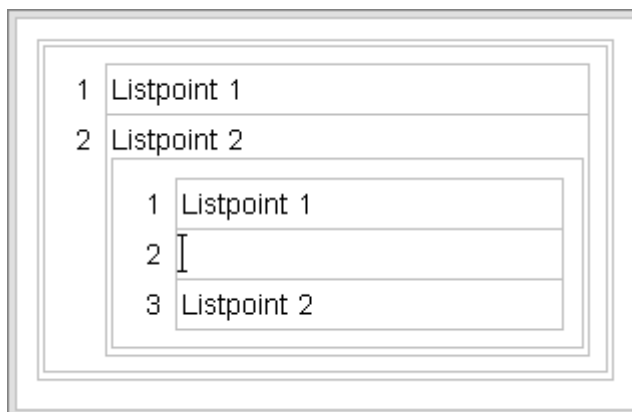


The numbering of the 1st list level is continued with the new bullet point.

**Figure 11-38: Divide nested list 1**

The cursor is then in the empty list point of the 1st numbered list level.

#### 11.5.6.4 Divide list within a nested list

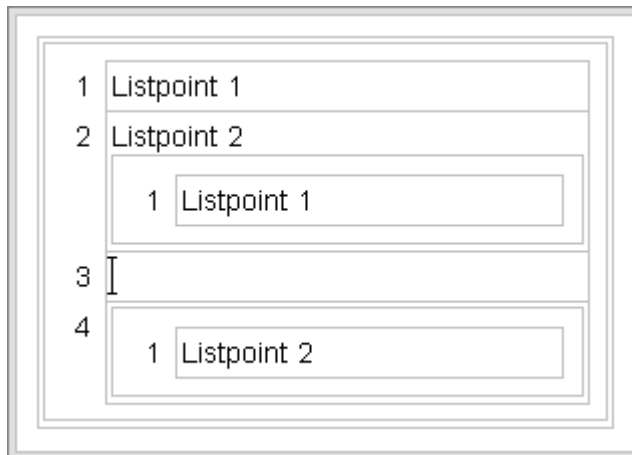


Press <RETURN> in an empty bullet point of the 2nd list level to divide the list at this point and move the empty nested bullet point to one level higher.

**Figure 11-39: Divide list within a nested list 1**







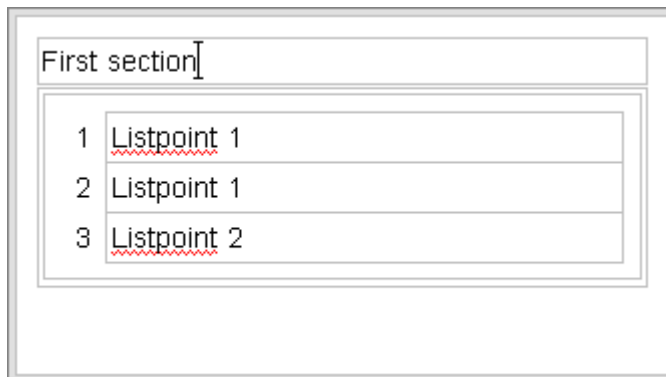
The numbering of the existing bullet points of the 2nd list level is reset accordingly. The numbering of the 1st list level is continued with the new bullet point and the second part of the nested list

**Figure 11-40: Divide list within a nested list 2**

The cursor is then in the empty list point of the 1st numbered list level.

## 11.5.7 Delete/merge lists

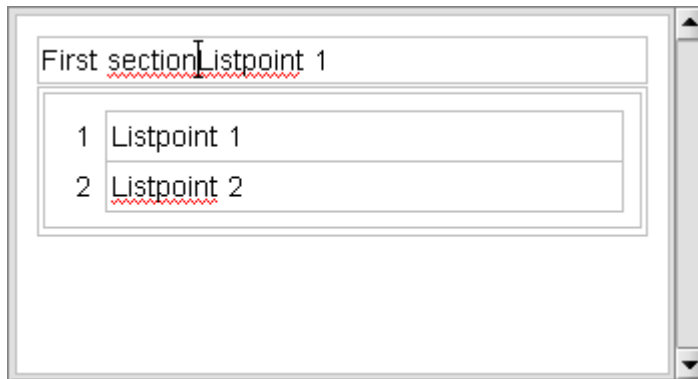
### 11.5.7.1 Merge a preceding section with the first list point



Press <DEL> at the end of the preceding section to merge the preceding section and the list to form one section.

**Figure 11-41: Merge list with preceding section 1**

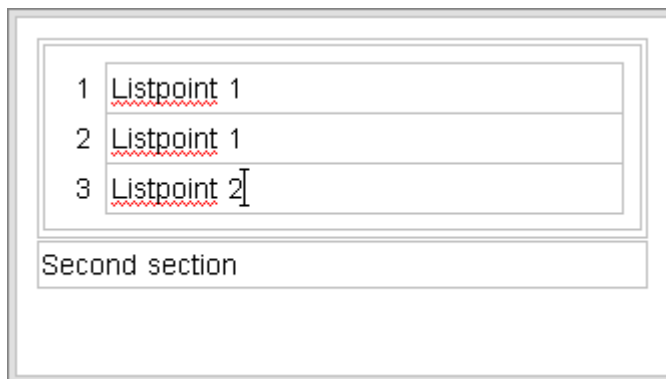




**Figure 11-42: Merge list with preceding section 2**

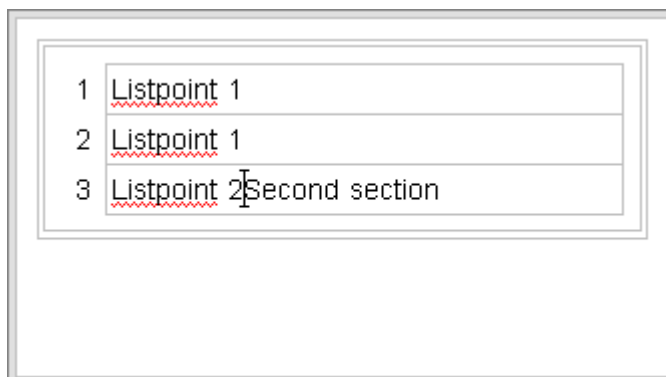
The cursor is then between the original section and list text.

### 11.5.7.2 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 11-43: Merge list with follow section 1**

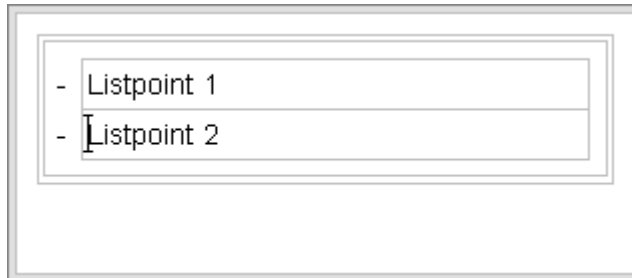


**Figure 11-44: Merge list with follow section 2**

The cursor is between the original list and section text.

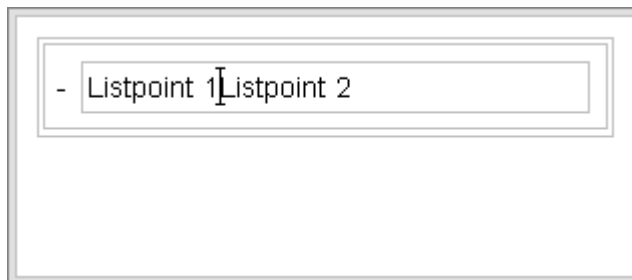


## 11.5.7.3 Merging two list points



Press <BACKSPACE> at the beginning of the following bullet point to merge it with the preceding bullet point.

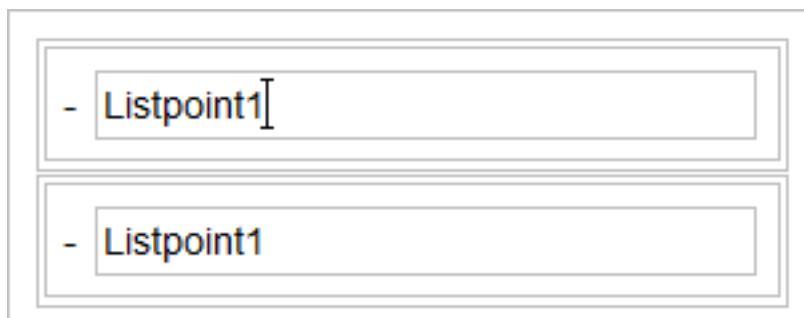
**Figure 11-45: Merge two list points 1**



**Figure 11-46: Merge two list points 2**

The cursor is located between the original list text of the preceding and of the following bullet point.

## 11.5.7.4 Merging two lists



If the two lists are each in a separate section, press <DEL> at the end of the first list to merge both lists in one list item.

**Figure 11-47: Merge two lists 1**





Then press <RETURN> to turn the two lists into one separate list item.

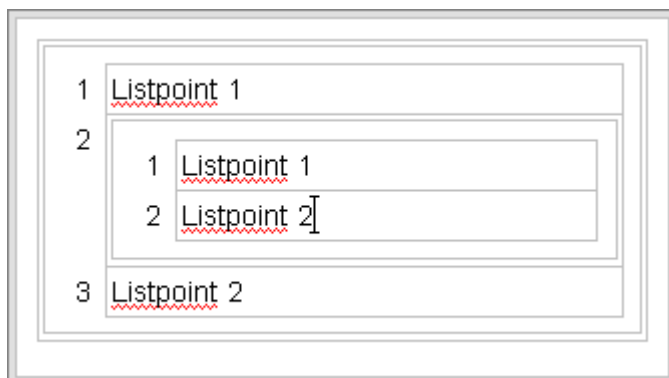
Figure 11-48: Merge two lists 1



Figure 11-49: Merge two lists 1

The cursor is then at the start of the first bullet point of the original second list.

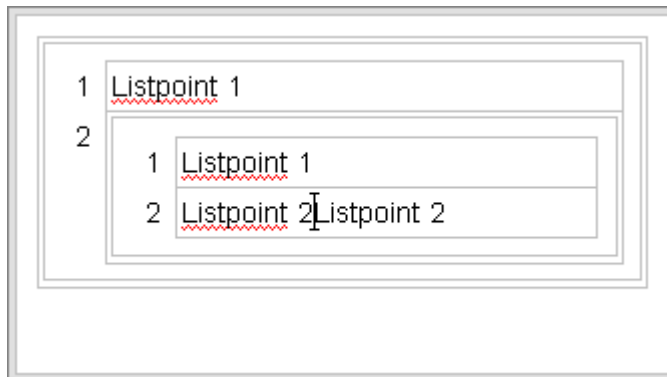
#### 11.5.7.5 Merge the last bullet point of an inner list with the following bullet point



Press <DEL> at the end of the last bullet point of the 2nd level to write the list text of the 1st level behind the text of the preceding bullet point of the 2nd level.

Figure 11-50: Merge inner list with the following bullet point 1





**Figure 11-51: Merge inner list with the following bullet point 2**

The cursor is located between the original list text of the 1st and 2nd level.

#### 11.5.7.6 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 11-52: Delete empty bullet point 1**



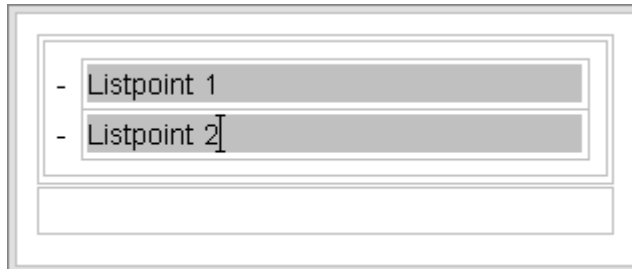
**Figure 11-53: Delete empty bullet point 2**

The cursor is then at the end of the preceding bullet point.



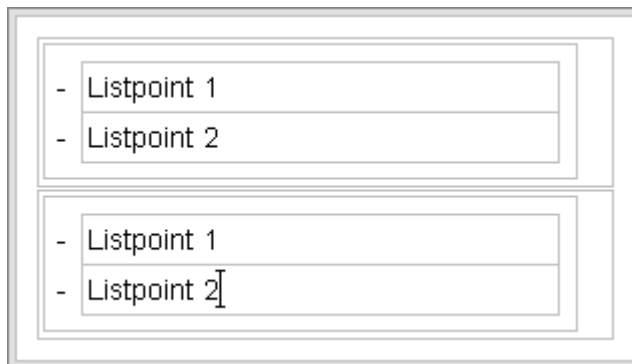
## 11.5.8 Copy/Move lists

### 11.5.8.1 Copy complete list



To copy the whole list, highlight it and press <CTRL> + <C>. Then position the cursor in an empty section and press <CTRL> + <V>.

**Figure 11-54: Copy list 1**

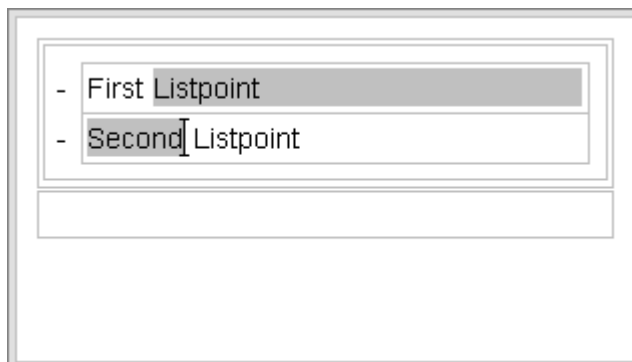


The complete list is pasted in the empty section as a new list

**Figure 11-55: Copy list 2**

The cursor is then at the end of the last bullet point.

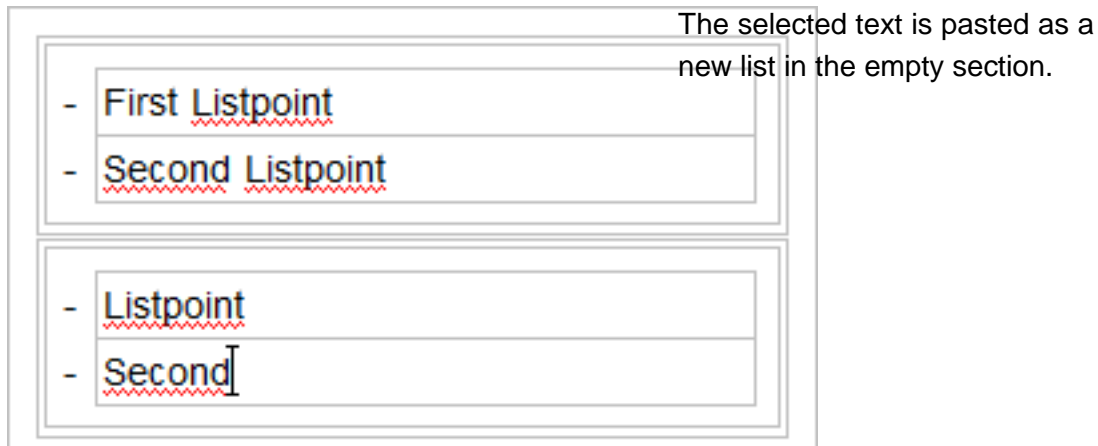
### 11.5.8.2 Copy partially highlighted list



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 11-56: Partially copy list 1**



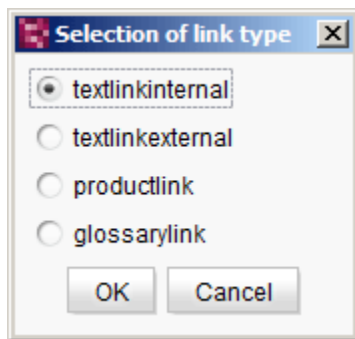


**Figure 11-57: Partially copy list 2**

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.

### 11.5.9 Insert links

If the [LINK](#) icon is used to insert a link at the cursor's current position, a window opens for selecting the link type.



**Figure 11-58: 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 link is analogous to the description of the "Link Input" input component (see section 10.8, page 285).




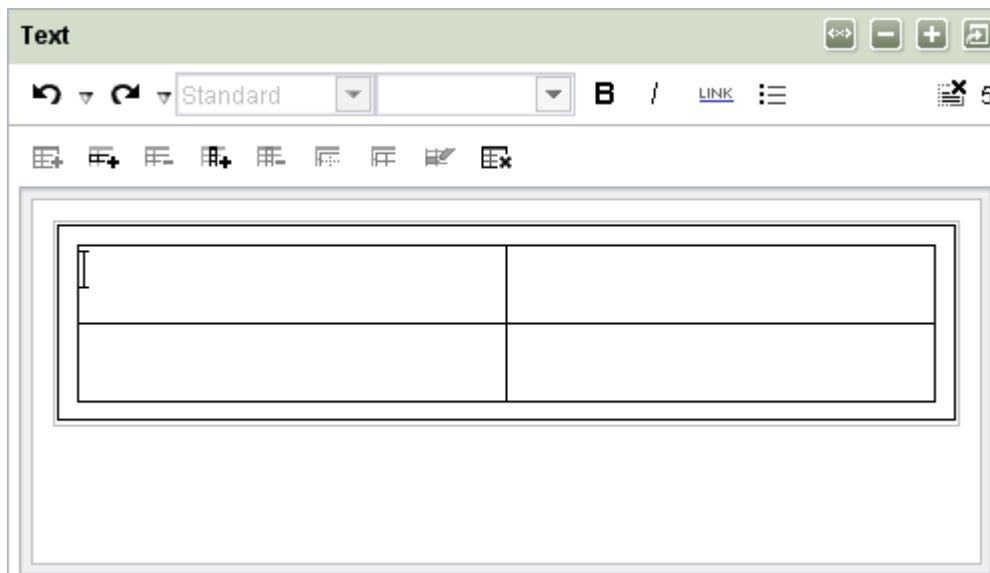
### 11.5.10 Integrate tables in the rich text editor

In rich text editor, tables can also be integrated in the text flow (so-called "inline tables"). The tables provide a variety of layout options, down to the cell level.



*Whether the inline function is available in the rich text editor or not depends on the template developer's default settings. If the  tool bar is available in the rich text editor, the Inline Tables function is also available.*

To insert a table into the rich text editor, the cursor is placed at the position within the rich text editor at which the table is to be inserted. Click the  icon to insert a new table. If more than one template is available they will be displayed below the icon. If only one template is available the table will be created on the basis of the template in the rich text editor:



**Figure 11-59: Table inserted**

The number of rows and columns and the formatting of the inserted table depend on the template developer's settings.


#### 11.5.10.1 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.

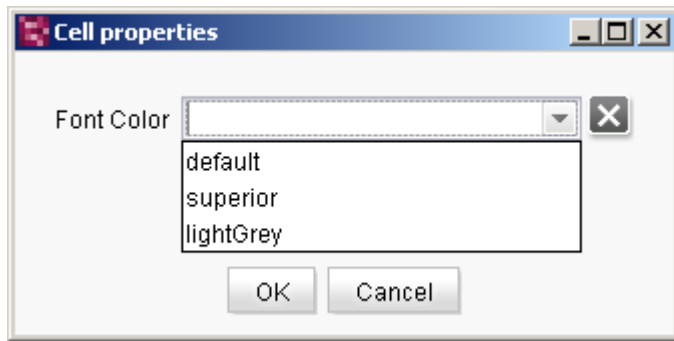
 Delete Column: click this icon to delete from the table the whole column including contents, in which the text cursor is positioned. As many columns as necessary can be deleted, until the minimum number preset by the developer is reached. The icon is then disabled, and no further columns can be deleted.

 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 11-60: Example: Table format template – Cell properties**

The required property can be selected from the combo box and saved with the **OK** button. Click the  icon 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.

#### 11.5.10.2 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 rich text editor. In this case the "Cell Properties" button in the top part of the rich text editor is active:



**Figure 11-61: "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 11-62: 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 colored marking. You can manually overwrite these values or accept them for the respective cell by clicking the   icon. 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 desired properties can be selected from the input components and saved with the **OK** button. Click  to remove the property again later.

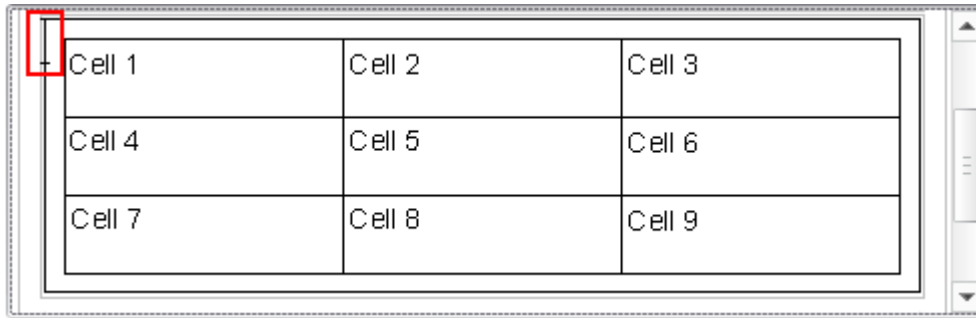


*If invalid values are used for the background color (e.g. #ZZZZZZ), the cell concerned is displayed with a white background color in the relevant cell in the rich text editor and in the subsequent output.*

### 11.5.10.3 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):

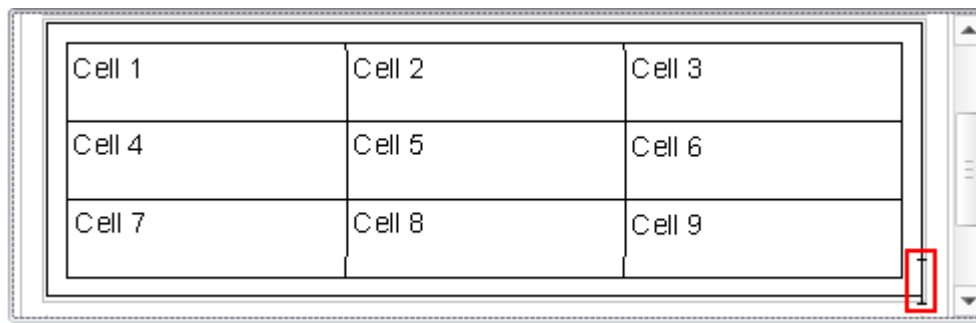




**Figure 11-63: Insert text section in front of inline table**

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 11-64: Insert text section after inline table**

The "Enter" key is then pressed to add a new text section after the table.

#### 11.5.10.4 Keyboard control of inline tables

The cursor keys ← → ↑ ↓ can be used to navigate the cells of inline tables (see also section 11.9.1, page 365). 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 in the same way as with the cursor key ←.



### 11.5.10.5 Further formatting options

The editor can use all format and link templates available in the rich text 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" rich text editors.

## 11.6 Working with the table input form

The table input form is intended 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.

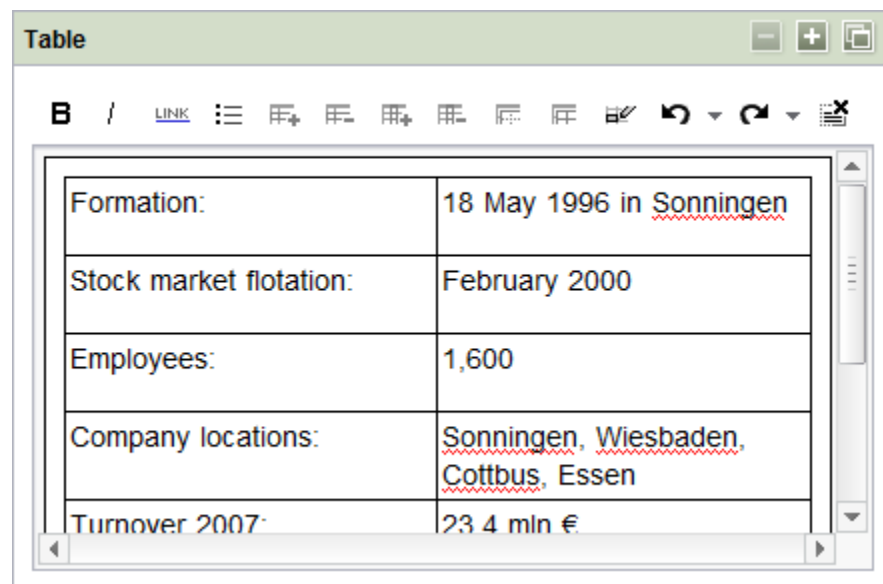


Figure 11-65: DOM Table input component



*Additional formatting options and the integration of tables into continuous text is provided by the so-called "inline tables". For more information see section 11.5.10, page 343.*

### 11.6.1 Icon bar of a table

The standard functions that can be opened via the icon bar have already been described in the Tables section under Standard Input Components. See section 10.4, page 277.



## 11.6.2 Context menus in tables

### 11.6.2.1 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.

### 11.6.2.2 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.

### 11.6.2.3 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 in the cell.

### 11.6.2.4 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 section 10.8 page 285.

**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.

## 11.6.3 Checking spelling in tables

Checking spelling in the Table input form is similar to the spelling check in the rich text editor (see section 11.5.3, page 324).



#### 11.6.4 Cell properties

The template developer specifies which cell properties can be changed. This could include, for example, text alignment, background color or number formats.

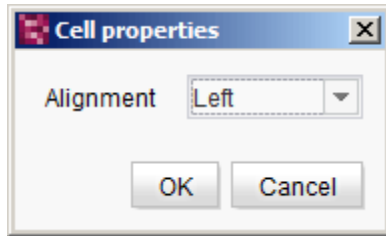



Figure 11-66: Cell properties



## 11.7 Selection dialogs

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, for example, images 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

- Link-sensitive graphic (section 10.17, page 300)
- Dataset selection (section 10.10, page 294),
- Reference selection (section 10.7, page 282),
- and in links whose configuration enables a reference to be selected (see also *FirstSpirit Manual for Developers*) (section 10.8, page 285)

a dialog opens which can be used to easily find and select the objects of the respective store. At the same time, only objects

- that 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 permissions configuration takes effect, as described in section 13.1.4.3, page 418.



*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, see section 11.8, page 362.*

### 11.7.1 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 (an image, 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.





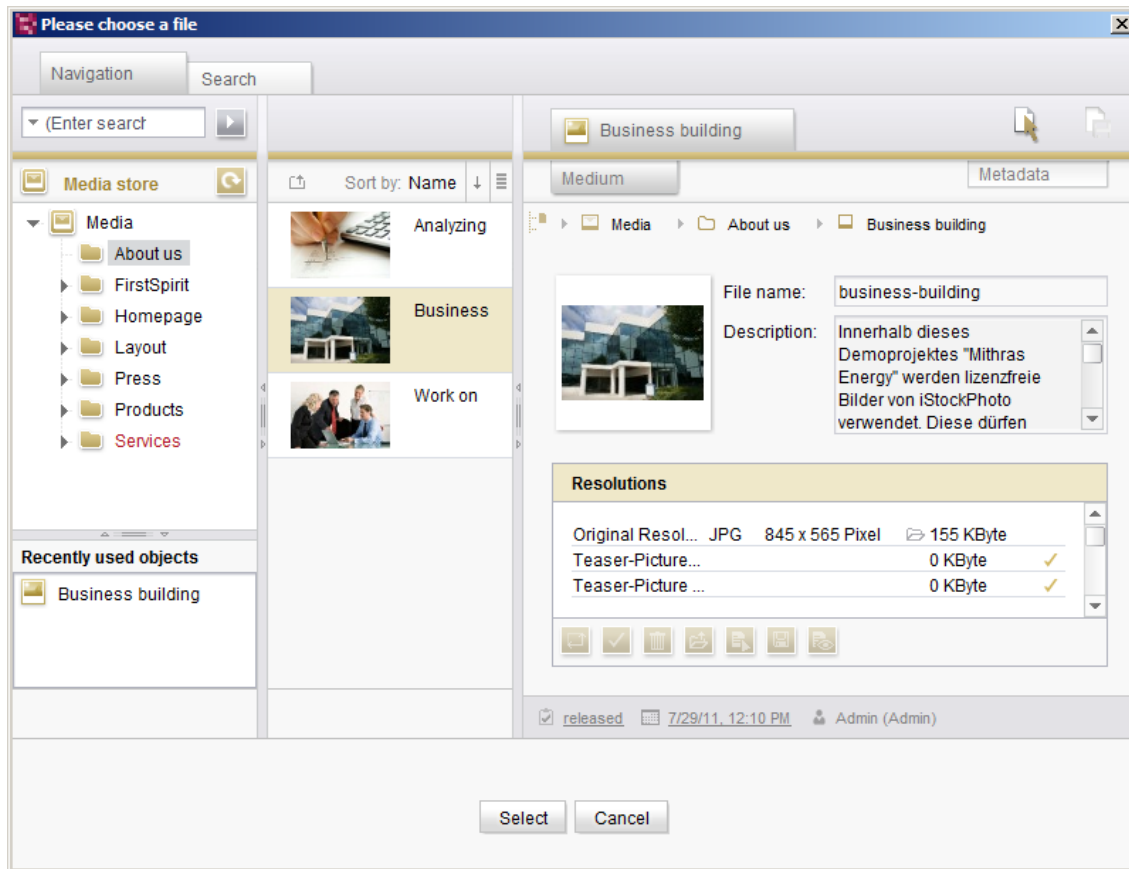



Figure 11-67: Image selection – Navigation tab

#### 11.7.1.1 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 is also included in the search, e.g. texts in PDF files. If the search is started with  or the "Enter" key, the view of the dialog automatically switches to the Search tab (see section 11.7.2, page 355). If a folder was selected in the tree structure (see section 11.7.1.2 page 352), it is adopted as the starting point for the search.

#### 11.7.1.2 Tree structure

The tree structure of the respective store can be seen in the left-hand column. Here the user can navigate directly to a folder or to an object within the folder.



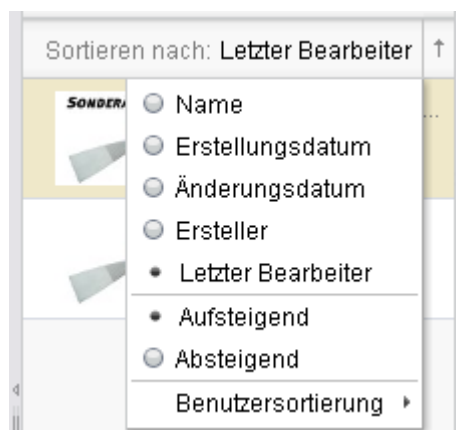
**Recently used objects:** Underneath the tree structure, there is a list of recently used objects for the user in question, making it easier to find objects that are used frequently.

### 11.7.1.3 Results list

Editorial content is also included in the search, e.g. texts in PDF files. Click on a folder in the left-hand column to display a list of the folders and objects available in this folder that may be selected, according to the specifications in the input component, with their display or reference names (see section 3.1.5.2, page 55, option "Display reference names in tree") and their thumbnails. Images are displayed with a miniaturized preview of the image; for all other file types, the corresponding thumbnails for the file format are displayed (e.g. folder or PDF icons).

Results list

**Sort by:** Click this area to open a list of criteria by which the results list can be sorted:



**Figure 11-68: Definition of sort criteria**

**Name:** Display or reference name of the object (see section 3.1.5.2, page 55, option "Display reference names in tree").

**Creation date:** Date and time at which the object was created in JavaClient.

**Change date:** Date and time at which the object was last edited.

**Creator:** Name of the user who created the object.

**Last editor:** 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 editor:




**Figure 11-69: Sorting by "Last editor"**

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).

#### 11.7.1.4 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 section 6.4 from page 188, for menu levels in the Site Store, see section 7.4 from page 238).

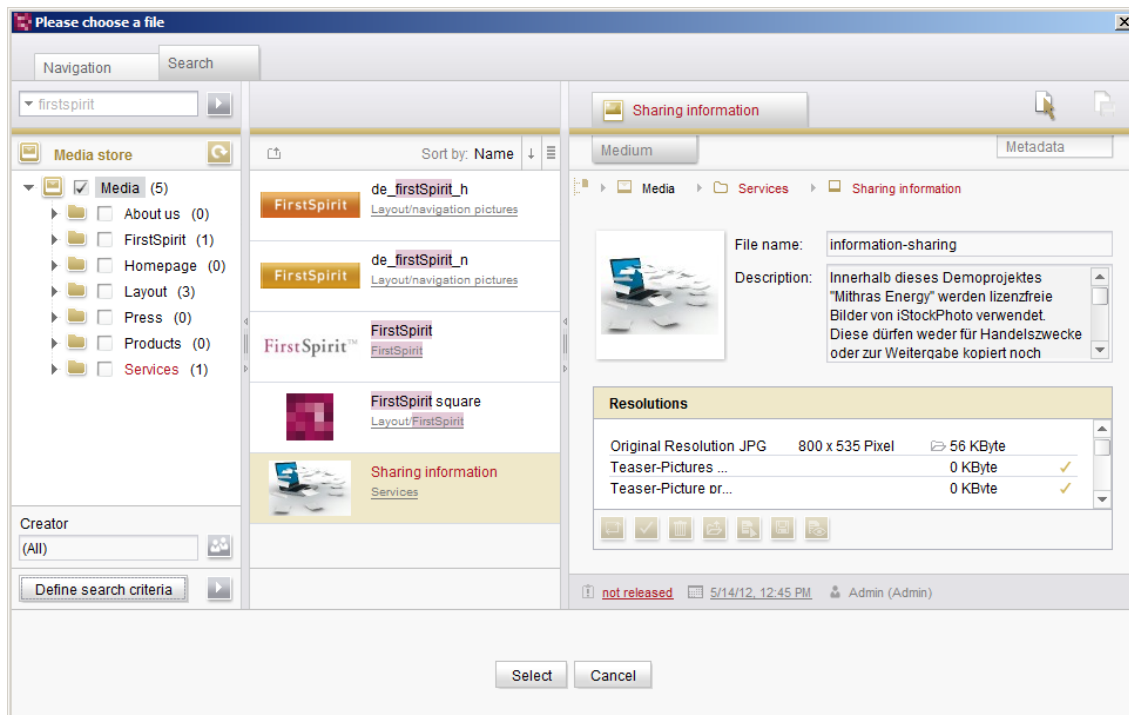
**At object level:** If an object is selected in the middle column, the details of the object including all tabs are displayed in the right-hand column. The settings which are also available in the respective store can be made for the object (for media, see e.g. section 6.5 from page 190, for files see section 6.7 from page 211 and for page references, see section 7.5 from page 246).

Editing and saving can be done using the familiar  and  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.



## 11.7.2 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 11-70: Image selection – Search tab**

Here too the view of the Navigation tab is divided into three parts (see section 11.7.1, page 351). In addition to searching for a specific search term, search results can also be filtered (e.g. by editor or change period) and sorted (see section 11.7.2.2 page 358).

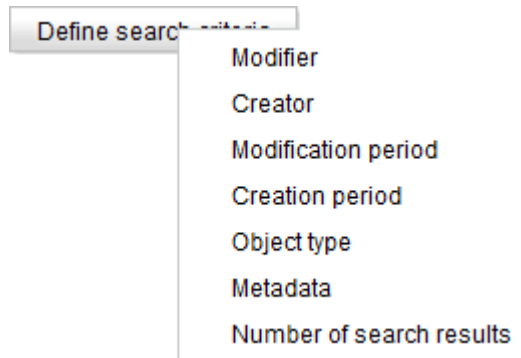
### 11.7.2.1 Tree structure

The tree structure of the store can be seen in the left-hand column. 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 selected. 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. Check marks in the list indicate which criteria are already active. The check marks are




disabled by clicking them again.



**Figure 11-71: 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 section 13.2.4, page 431).

**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 section 13.2.4, page 431).

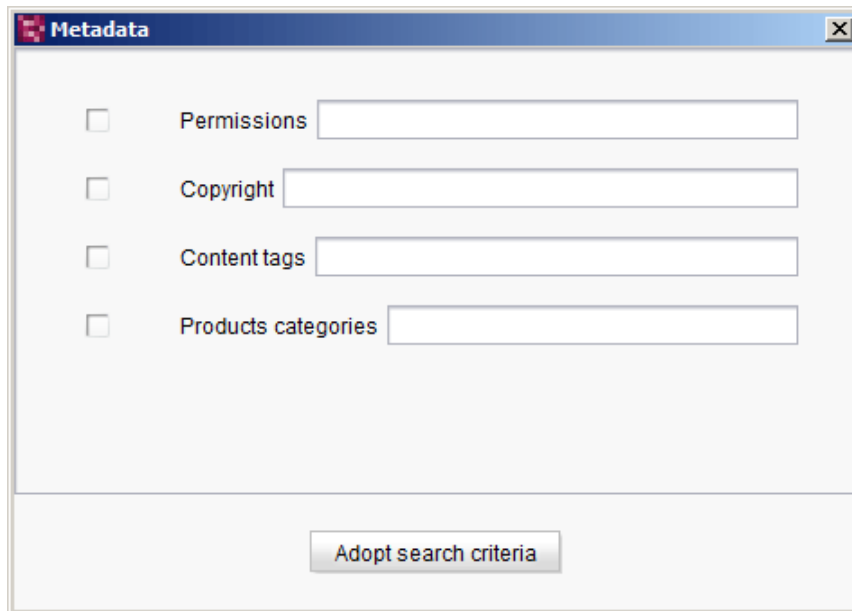
**Modification period:** This combo box can be used to limit the search to objects changed within a certain period. If the "user defined" option is selected, the calendar icons can be used to individually define the period.

**Creation period:** This combo box 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.

**Object type:** This combo box can be used to limit the search to certain types of objects (images, 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 is defined. Another dialog opens, in which criteria and values for can be defined for the metadata search:

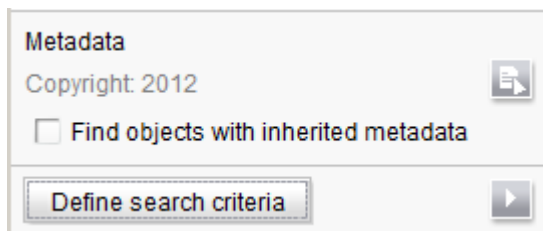




**Figure 11-72: Metadata search example**

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 11-72 are displayed in the search criteria area as follows:



**Figure 11-73: "Metadata" search filter**

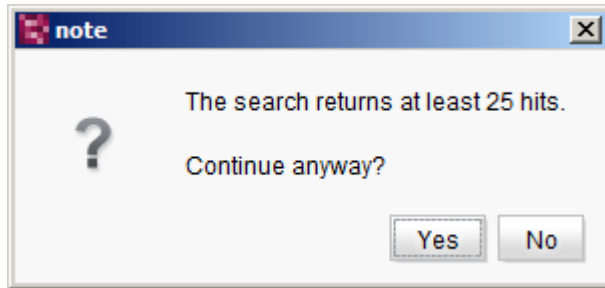
Here the **Find objects with inherited metadata** option can be used to also define whether objects whose metadata was only inherited are also to be included in the search. If a checkbox is not selected, only search results which explicitly contain the value given for the search criterion in the metadata 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 section 11.3, page 309 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 11-74: 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 canceled 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.

### 11.7.2.2 Results list

The results of the search are displayed in the middle column, analogous to those in the Navigation tab (section 11.7.1.3, page 353). 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 color in the results list and the path is also displayed. The sort options are also analogous to those in the Navigation tab.

### 11.7.2.3 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. section 6.5 from page 190, for files see section 6.7 from page 211 and



for page references, see section 7.5 from page 246).

Editing and saving can be done using the familiar  and  icons of the icon bar. The **Select** button is used to copy the object into the input component and then the dialog closes again.

### 11.7.3 Navigation using the keyboard

Apart from the keyboard shortcuts described in section 11.9 from page 363, 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 content 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 level up
Double-click	= on folders in the tree structure: Open the next folder level = on folders in the results list: Display the folder content in the results list = <b>on elements in the results list:</b> Copy the element into the input component; the selection dialog closes





### 11.7.4 Multiple selection in selection dialogs

Depending on the template developer's configuration, multiple selection may also be possible for the "List creation" input form.

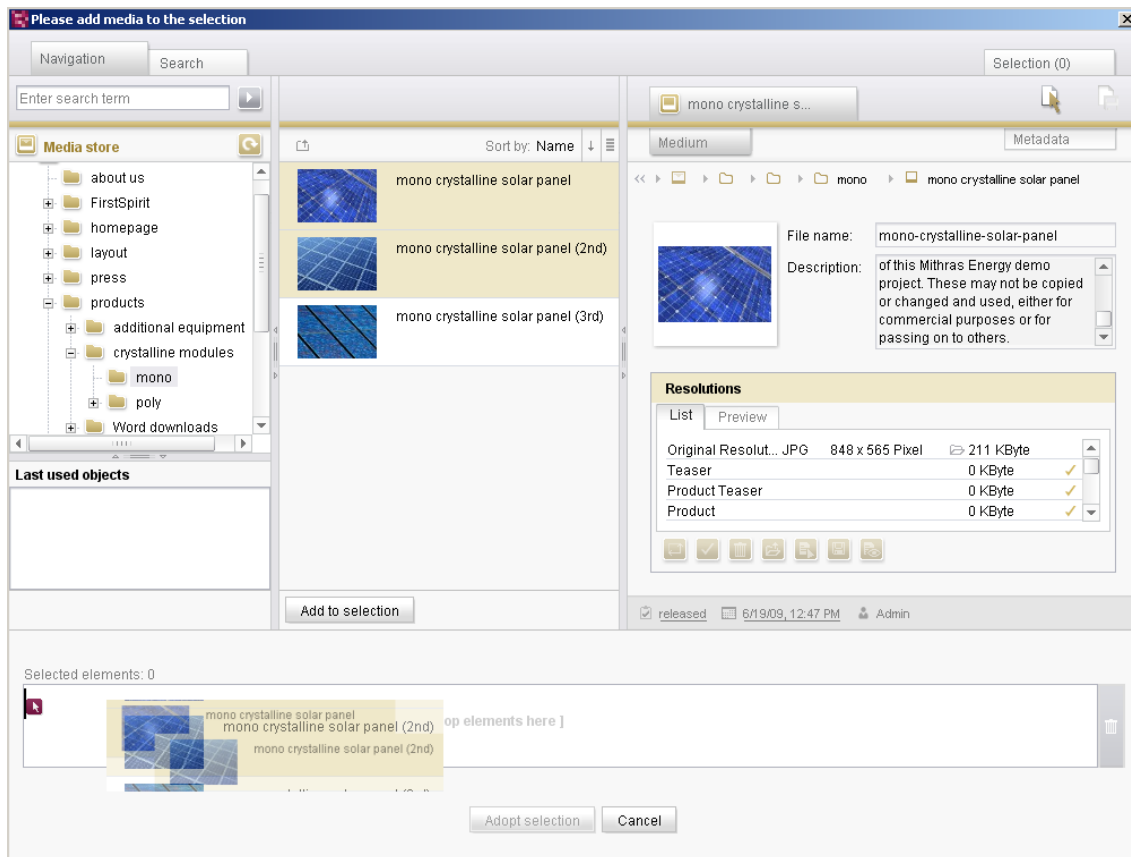


Figure 11-75: Multiple selection in selection dialogs

The difference to simple selection is that there is a list of selected elements. The elements can either be included in the list of selected elements by drag-and-drop or via the **Add to selection** button. Multiple selection is possible by pressing the <CTRL> and <SHIFT> key. Depending on the template developer's default settings, folders can also be selected.

The selection of all selected elements can then be directly copied into the input component ("Adopt Selection") or can be further edited first in the "Selection" tab.

**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:



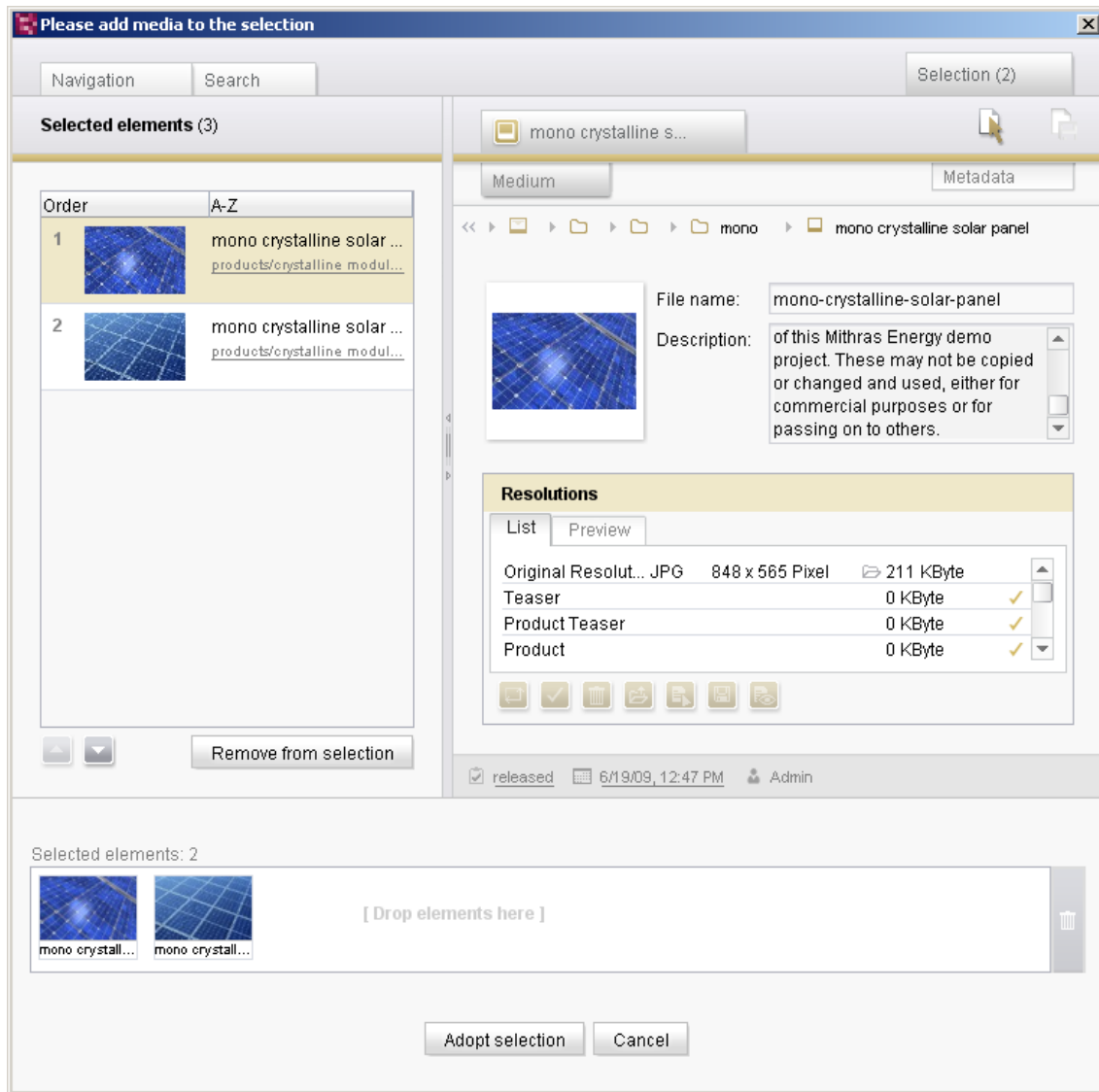


Figure 11-76: Selection tab: Edit selected media

**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 form 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-and-drop to place them in the drop zone:

**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 image can be requested in the selected resolution.

The list of selected media can be exited again by clicking the "Navigation" or "Search" tab and other media can be selected.

## 11.8 Working with media galleries

FirstSpirit provides an option for selecting images from the Media Store and displaying them in a gallery view. The same image can be used in several galleries, each with different description texts.

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 images of the gallery can be page d through in an order definable by the editor.
- **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. 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).

The creation and maintenance of galleries requires several steps in different stores. The following stores are relevant for the editorial maintenance of media galleries:

- **Media Store:** The images to be subsequently displayed in a gallery are uploaded here and can be maintained as usual.



- **Content Store:** All information on galleries is managed in a corresponding data source. Here, for example, 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.
- **Page Store:** Within the Page Store, individual galleries can be integrated on a page or gallery overviews can be generated. In addition, the Page Store can also be used to maintain gallery data.

## 11.9 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 tool bar, 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 tool bar:

<ALT> + ←/→	= Go back/forwards to objects that have already been selected once during the current session
<F5>	= Refresh
<CTRL> + <E>	= Enable/disable Editing mode
<CTRL> + <S>	= "Save" button on the client tool bar
<CTRL> + <P>	= "Preview" button
<CTRL> + <N>	= "New" button on the client tool bar
<Del>	= Delete
<CTRL> + <T>	= Open task list
<F1>	= Open online help

### Context menu items

Application key 	= Open the context menu of the respective node
<CTRL> + <SHIFT> + <N>	= Create new folder
<CTRL> + <Z>	= Undo (back to the last saved status of the node)



or <CTRL> + <SHIFT> + <Z>	=	with confirmation prompt)
<CTRL> + <X>	=	Cut
<CTRL> + <C>	=	Copy
<CTRL> + <V>	=	Paste
<CTRL> + <D>	=	Create document group (Site Store only)
<F9>	=	Rename
<CTRL> + <U>	=	Show usages (if only one usage exists, the focus switches to that usage)
<CTRL> + <SHIFT> + <E>	=	Cancel editing, unsaved data is lost
<CTRL> + <H>	=	Open version history on the node in question
<CTRL> + <R>	=	Show dependent objects of the node (reference graph)
<CTRL> + <W>	=	Close the current workspace
<CTRL> + <R>	=	Display dependencies (Reference graph)
<User-defined>	=	Depending on the template developer's settings, user-defined keyboard shortcuts may be available for starting and switching workflows (section 12, page 390).

### Functions in the extended workspace

<ALT> + <A>	=	Navigate to first workspace
<ALT> + <Q>	=	Navigate to previous workspace
<ALT> + <W>	=	Navigate to next workspace
<ALT> + <S>	=	Navigate to last workspace
<ALT> + <ESC>	=	Close active workspace
<ALT> + <TAB>	=	Activate right-hand workspace from currently active workspace
<ALT> + <Minus>	=	Zoom out
<ALT> + <Plus>	=	Zoom in
<ALT> + <0>	=	Reset zoom

### Selecting and moving objects

<CTRL> + <left-click>	=	Multiple selection of individual elements
<SHIFT> + <left-click>	=	Multiple selection of multiple elements
<SHIFT> + <cursor>	=	Multiple selection of multiple elements/expand selection up/down
<CTRL> + <A>	=	Select all elements
<Drag with mouse>	=	Move objects
<SHIFT> + <drag with mouse>	=	Copy objects

### Other functions

<CTRL> + <Z>	=	Undo last action
<CTRL> + <F>	=	Find
<F3>	=	Find next
<SHIFT> + <F3>	=	Find previous



<ALT> + <P>	= Show node info
<F10>	= Activate main menu level, navigate with cursor keys
<F11>	= Enable/disable full screen mode for the current window
<F12>	= Open the special characters table
<SHIFT> + <CTRL> + <drag with mouse>	= Create section reference (Page Store only)
<CTRL> + <L>	= Show line numbering (Template Store only)

### Keyboard shortcuts in the "Edit URL" dialog box (Chapter 9.3 page 260)

<Enter>	= One line forward/down
↓	= One line forward/down
<Enter> + <SHIFT>	= One line back/up
↑	= One line back/up
<Enter> + <CTRL>	= Input in the "URL" field in the selected line, text input starts to the right of the existing text

### 11.9.1 Keyboard shortcuts in the rich text editor

<ENTER>	= Inserts a new section
←	= Cursor one position to the left
→	= Cursor one position to the right
↑	= Cursor one position up
↓	= Cursor one position 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 row
<END>	= Go to the last character in the current row
<ALT> + <HOME>	= Go to the first character in the section
<ALT> + <END>	= Go to the last character in 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> + ←	= Selects the text from the current position to the preceding character
<SHIFT> + →	= Selects the text from the current position to the next 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 below
<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 to the next preceding space character
<SHIFT> + <CTRL> + →	= Highlights the text from the current position 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	= Highlights the current word
Triple-click	= Highlights the current row
Quadruple-click	= Highlights 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>	= Deletes a word from the cursor position to the end of the word, including a following punctuation mark
<CTRL> + <Backspace>	= Deletes a word from the cursor position to the beginning of the word
<CTRL> + <V>	= Pastes the content of the clipboard at the cursor's position
<CTRL> + <F>	= Opens a window to 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>	= Restores changes that have been undone (redo)
<CTRL> + <Y>	= Restores changes that have been undone (redo)
<Page Up ↑>	= Pages up without moving the cursor
<Page Down ↓>	= Pages down without moving the cursor
<CTRL> + <B>	= Bold formatting
<CTRL> + <I>	= Italic formatting
<CTRL> + <SHIFT> + <R>	= Dialog for inserting a link
<CTRL> + <SHIFT> + <L>	= Dialog for inserting a lists
<CTRL> + <SHIFT> + <P>	= Focus on selection list with section formats, selection of the desired section format template using cursor keys
<CTRL> + <SHIFT> + <T>	= Focus on selection list with character formats, selection of the desired section format template using cursor keys

For further information about the keyboard shortcuts that can be used in tables in the rich text



editor see section 11.5.10.4, page 347.

## 11.9.2 Keyboard shortcuts for DOM table

<ENTER>	=	Inserts line break at the current position
←	=	Moves cursor one position to the left within a cell or to the next cell at the left
→	=	Moves cursor one position to the right within a cell or to the next cell at the right
↑	=	Moves cursor one position up within a cell or to the next cell up
↓	=	Moves cursor one position down within a cell or to the next 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 in the cell
<ALT> + <END>	=	Go to the last character in the cell
<SHIFT> + ←	=	Highlights the text from the current position up to the preceding character. Each additional click on ← expands the selection one more character to the left. If the cursor is in front of the 1st character of a cell, clicking ← will select the current cell as well as the next cell to the left. If there is not any cell to the left of the current cell, the current 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 → expands the selection one more character to the right. If the cursor is after the last character of a cell clicking → will select the current cell as well as the next cell to the right. If there is not any cell to the right of the current cell, the current 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 row above. If the cursor is in the top row of a cell, the current cell and the cell above it are selected.
<SHIFT> + ↓	=	Highlights the text from the current position up to the same position in the next row. Each additional click on ↓ the selection will be expanded one line beneath. If the cursor is in the last row of a cell, the current cell and the cell under it are 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 to the next preceding space





	character
<SHIFT> + <CTRL> + →	= Highlights the text from the current position 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>	= Highlights the whole content of the DOM table
<b>Double-click</b>	= <b>Highlights the current word</b>
Triple-click	= Selects the current row in a cell
Quadruple-click	= Selects the current cell
<CTRL> + <C>	= <b>Copies the highlighted content into the clipboard</b>
<CTRL> + <X>	= <b>Cuts the highlighted content and copies it into the clipboard</b>
<CTRL> + <Del>	= Deletes a word from the cursor position to the end of the word, including a following punctuation mark
<CTRL> + <Backspace>	= Deletes a word from the cursor position to the beginning of the word
<CTRL> + <V>	= <b>Pastes the content of the clipboard at the cursor's position</b>
<CTRL> + <F>	= Opens a window to find text in the table
<CTRL> + <R>	= Opens a window to find and replace text in the 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>	= Restores changes that have been undone (redo)
<CTRL> + <Y>	= Restores changes that have been undone (redo)
<Page Up ↑>	= Pages up without moving the cursor
<Page Down ↓>	= Pages 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.



## 11.10 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.

### 11.10.1 Terms and concepts

#### 11.10.1.1 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, for example, pages are compiled from individual sections and are combined in the Site Store to form a 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 section 12.3, page 393). At the technical level, when a change is released a specific version of an object is labeled as being "released".

#### 11.10.1.2 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.

### 11.10.1.3 Supported objects

The version history is available for the following stores and objects:

- In the Page Store on pages (see section 11.10.3, page 380)
- In the Content Store on datasets (see section 11.10.4, page 382)
- In the Media Store on media (see section 11.10.5, page 383)
- In the Site Store on page references (see section 11.10.6, page 384)
- In the Template Store (see *FirstSpirit Manual for Developers*)



### 11.10.2 Version history functions

Revision	Date	Change on	Editor	Comment	Attributes	Child list	Content	Metadata
20512	Sep 24, 2012 12:33:31 PM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
20468	Sep 19, 2012 11:49:41 AM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20467	Sep 19, 2012 11:01:06 AM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20460	Sep 17, 2012 11:32:30 AM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20459	Sep 17, 2012 11:11:22 AM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20458	Sep 17, 2012 10:39:02 AM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20457	Sep 17, 2012 9:23:33 AM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20446	Sep 12, 2012 10:51:04 AM	Mithras Homepage	Admin (Admin)	Release by server	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20438	Sep 5, 2012 12:15:46 PM	Mithras Homepage	shoebbel (Sandra Höbb...)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20437	Sep 5, 2012 12:15:15 PM	Mithras Homepage	shoebbel (Sandra Höbb...)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20436	Sep 5, 2012 12:14:47 PM	Mithras Homepage	shoebbel (Sandra Höbb...)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20435	Sep 5, 2012 12:09:30 PM	Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Selection**

1st revision: 20459 (Sep 17, 2012 11:11:22 AM) [Restore] [Display]

2nd revision (Ctrl): [ ] [Compare]

**Options**

- Show changes to ChildElements
- Show changes to Media
- Show changes to Templates
- Show hidden Revisions
- Show partially archived revisions

**Figure 11-77: Version history in the Page Store**

The version history is divided into the following areas:

- Navigation (see section 11.10.2.1, page 372)
- Displaying the revisions for an object (see section 11.10.2.2, page 372)
- Restoring revisions (see section 11.10.2.3, page 374)
- Version comparison (selecting revisions) (see section 11.10.2.5, page 377)
- Options (see section 11.10.2.6, page 379)



*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" section.*



11.10.2.1 Navigation area

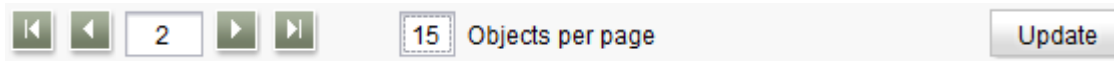


Figure 11-78: Version history – Navigation

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 page d 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 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).

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 the **Refresh** button to synchronize the task list of the FirstSpirit Server with the open revision view.

11.10.2.2 Display the revisions for an object

Revision	Date		Change on	Editor	Comment	Attributes	Child list	Content	Metadata
19635	May 2, 2012 10:18:55 AM		Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19630	May 2, 2012 9:25:28 AM		Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19622	Apr 30, 2012 11:52:49 AM		Mithras Homepage	Admin (Admin)	Save	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure 11-79: 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.





*If an archiving job has been performed (see FirstSpirit Manual for Administrators), only the revisions which can be restored are displayed in this list: i.e. at least the revisions of the current edit status and, if available, the last release status. The "Show hidden Revisions" option can be used to show elements of the version history whose content has been partially archived.*

**Revision:** Revision number of the displayed object (see section 11.10.1.2, page 369)

**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 "View/Preferred display language" setting (see section 3.1.5.2, page 55).

**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).



*If the project properties were configured accordingly, comments assigned by the editor are displayed here (see also Figure 3-41). 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 overview in section 12.4.3, page 404):



## 11.10.2.3 Restore revisions

In the bottom part of the page it is possible to restore a revision selected in the table view.

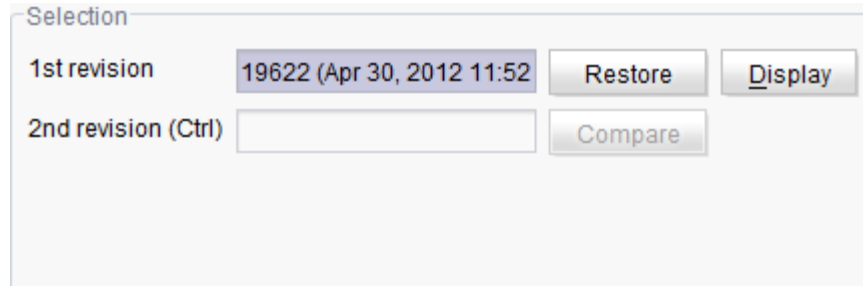


Figure 11-80: Version history – Restore revisions

If an older revision of an object has been highlighted in the table overview, click the **Restore** button to restore the selected revision of the object. The options for a restore action can be selected in the following pop-up window:

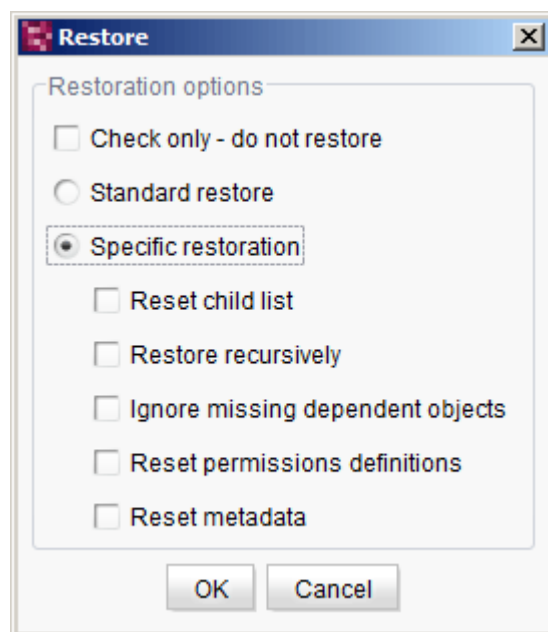


Figure 11-81: Restore – Options

**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 restoration" area, depending on the object.

**Specific restoration:** This option can be selected to manually adjust the standard restore options.

**Specific restoration – 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 restoration – Restore recursively:** If this option is selected, all the selected options are applied on all objects (children) located below the selected object.

**Specific restoration – 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 restoration – 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 restoration – 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 is 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 **Display 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 labeled with the comment "Restored". The "Restore" function is deactivated in the respective






current revision.

Revision	Date		Change on	Editor	Comment	Attributes	Child list	Content	Metadata
19903	Jun 4, 2012 11:01:40 AM		Mithras Homepage	Admin (Admin)	Restore revision 19630	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
19902	Jun 4, 2012 11:01:09 AM		Mithras Homepage	Admin (Admin)	Restore revision 19898	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
19899	Jun 4, 2012 11:00:54 AM		Mithras Homepage	Admin (Admin)	Workflow 'Freigabe Anfor...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure 11-82: 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 datasets (both the number and the editorial content) do not correspond to the content of the datasets at the time of the revision. This behavior applies to all dependencies within an object, in other words also page references and the dependent pages, for example. The relevant content should therefore be explicitly checked after being restored.*

#### 11.10.2.4 Display older revisions (historic preview)

Selection

1st revision

2nd revision (Ctrl)

Figure 11-83: Version history – Restore revisions

If an object is selected in the table view, click the **Display** button to display a preview of the selected revision of this object.



### 11.10.2.5 Version comparison – Select revisions

Changes to an object which have taken place from one revision number to another revision number can be displayed using difference visualization. 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.



*Revision comparison is only possible for two revisions of the original element. If the selected revisions are of different element types, comparison is not possible → the "Compare" button is disabled.*

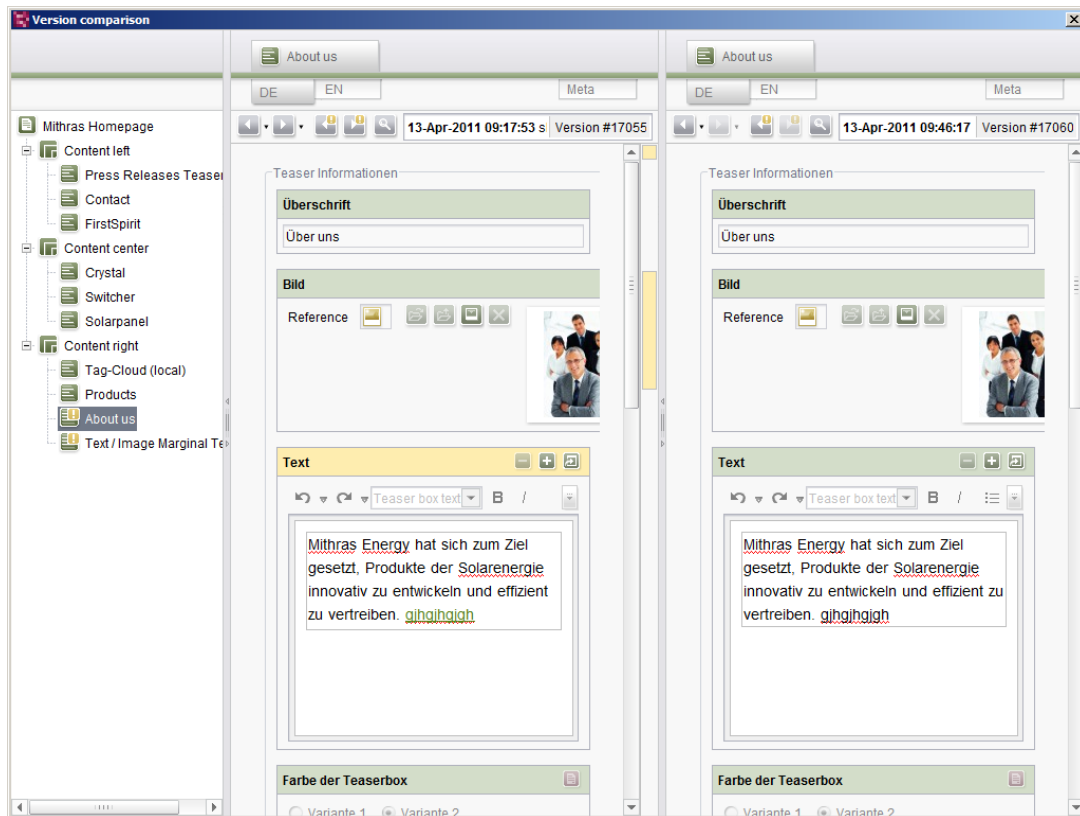
The two revisions are displayed in the lower part of the page in the fields **1st revision** and **2nd revision**.

Selection	
1st revision	19731 (May 16, 2012 9:06) [Restore] [Display]
2nd revision (Ctrl)	19592 (Apr 30, 2012 8:26:) [Compare]

**Figure 11-84: Version history – Select revisions for comparison**

When the **Compare** button is clicked the versions are compared and the dialog "Version comparison" opens.





**Figure 11-85: Version comparison (Page Store) dialog**

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.

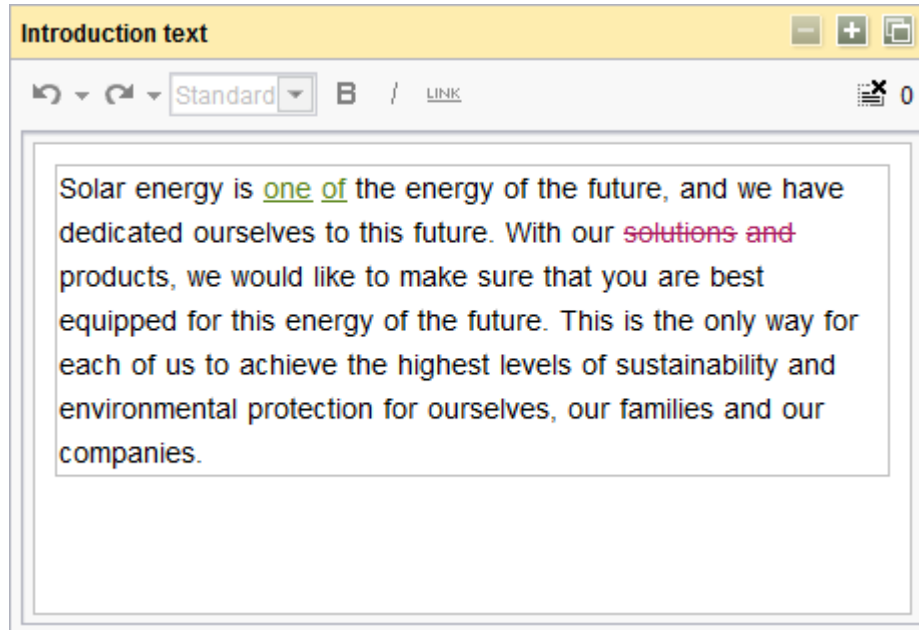
The selected revisions are displayed in the two columns on the right. The revision information such as version number, date, time, last editor and comment entered for the revision, are displayed at the top in the respective column. The Next and Back icons can be used to switch to the previous or next revision ("Revision selection") or the previous or next change ("Display changes") to the respective object.

Between the two right-hand window areas, a bar shows which input forms contain changes. Yellow marking indicates that data has been changed, red means that data has been removed and green that new values have been added. If the order of entries of the input form "List creation" has been changed for example, this will be shown with yellow marking, too. A tool tip displays the name of the input component concerned and it is possible to switch directly to it with a click.

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 is shown for an older comparison version.



**Figure 11-86: Changes (version comparison)**

By default, the changes within the rich text editor input form are marked separately; deleted (by default: highlighted red) and added content is displayed (by default: highlighted green).



*The version comparison serves only to compare two revisions. A revision may not be subsequently edited or changed. Therefore, content cannot be copied or changed from one version to another in the version comparison.*



*The default input form button is always hidden in a version comparison.*

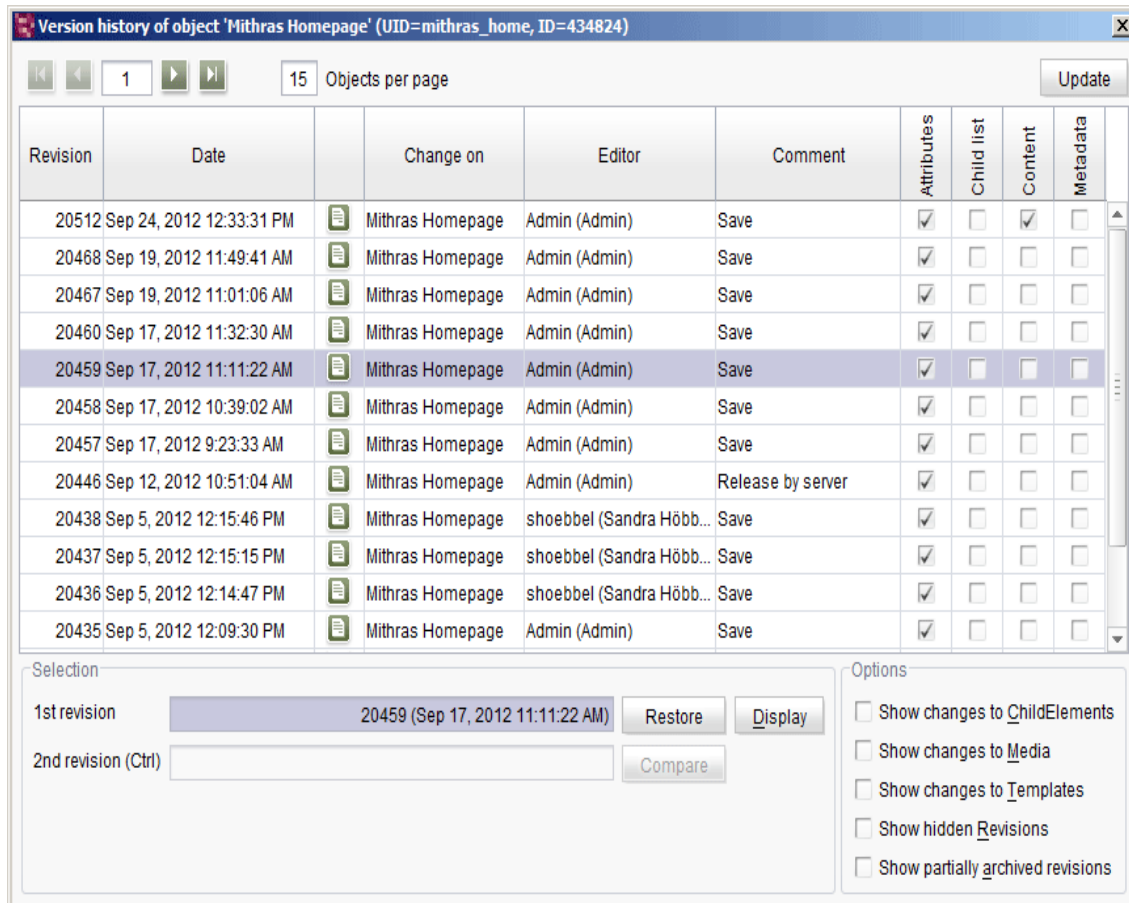
### 11.10.2.6 Options

Depending on the store and the object on which the version history is opened, further options of the version history are shown in the lower right-hand area of the window (see relevant overview in section 12.4.3, page 404):



### 11.10.3 Version history in the Page Store

Within the Page Store, version management is available **at page level**. It is opened via the context menu of the required page (see section 4.1.10, page 134). Click the "Version history" entry to open the dialog:



**Figure 11-87: Version history at page level**

The view of the version history can be adjusted in the top part of the window (see section 11.10.2.1, page 372).

The table lists the revisions of an object and, apart from the generally available information (see section 11.10.2.2, page 372), the following information is also displayed for the pages of the Page Store:

**Attributes:** If this checkbox is activated, the change which led to assignment of a new revision number involves a change to the page 's attributes (e.g. when a page 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 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 section 11.10.2.3, page 374. For details of comparing two revisions of a page, see section 11.10.2.4 page 376).

Options: 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 gray.

**Show changes to media:** If this checkbox is activated, changes to media that are referenced in this object are displayed.


**Show changes to templates:** If this checkbox is activated, changes to the object template are displayed.

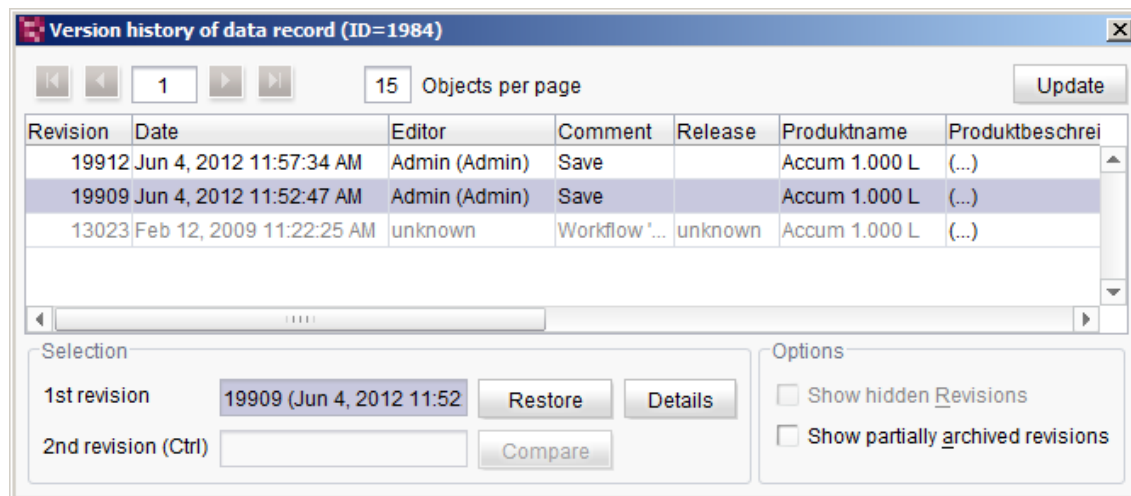
**Show hidden revisions:** If this checkbox is activated, additional internal system revisions of an object are displayed (if they exist). If an archiving job has been performed (see *FirstSpirit Manual for Administrators*), this checkbox can be used to show elements of the version history whose content 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 partially 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 gray. The relevant buttons can be used to perform the "Restore", "Display" and "Compare" functions on them.



### 11.10.4 Version history in the Content Store

Within the Content Store, the version history is available on **datasets**. The version history is called using the context menu of the required dataset (see section 5.2.5, page 158) or by clicking the  icon in the dataset's edit window.



**Figure 11-88: Version history at dataset level**


The view of the version history can be adjusted in the top part of the window (see section 11.10.2.1, page 372).

In addition to the generally available information (see section 11.10.2.2, page 372), the respective revision of all data entered for a dataset is listed here in table 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.

#### Selection:

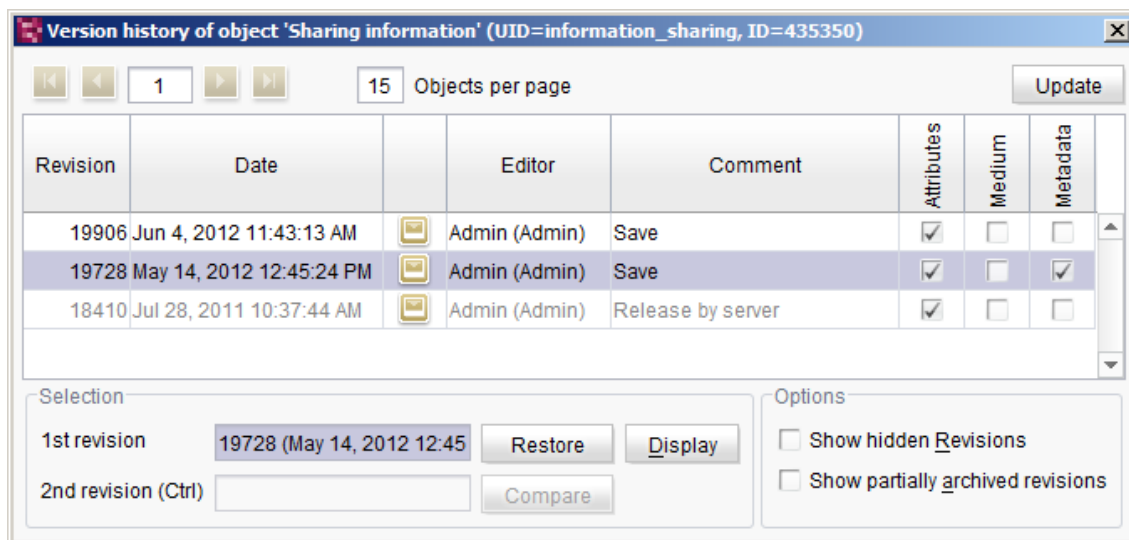
For information on restoring a revision, see section 11.10.2.3, page 374. For information on comparing two revisions of a dataset, see section 11.10.2.5, page 377.



 *New datasets created which have not yet been saved do not yet have a version history. The context menu and the icon are therefore hidden.*

### 11.10.5 Version history in the Media Store

Within the Media Store, version management is available at media level. It is opened via the context menu of the required medium (see section 6.1.10, page 182). Click the "Version history" entry to open the following dialog:



**Figure 11-89: Version history at media level**

The view of the version history can be adjusted in the top part of the window (see section 11.10.2.1, page 372).

The table lists the revisions of the medium and apart from the generally available information (see section 11.10.2.2, page 372) 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 image 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 section 11.10.2.3, page 374. For details of comparing two revisions of a page, see section 11.10.2.5 page 377).

Options: Other changes concerning the current object can be selected for display in the Options area.

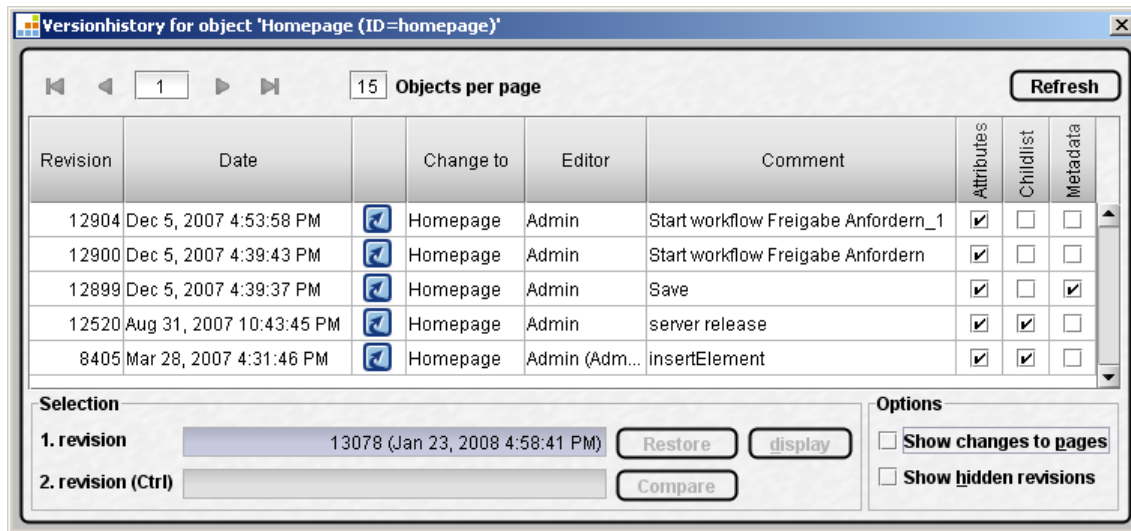
**Show hidden revisions:** If this checkbox is activated, additional internal system revisions of an object are displayed (if they exist). If an archiving job has been performed (see *FirstSpirit Manual for Administrators*), this checkbox can be used to show elements of the version history whose content 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 partially 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 gray.

### 11.10.6 Version history in the Site Store

Within the Site Store, version management is available at the **level of a page reference**. It is opened via the context menu of the required page reference (see section 7.1.10, page 231). Click the "Version history" entry to open the following dialog:





**Figure 11-90: Version history at page reference level**

The view of the version history can be adjusted in the top part of the window (see section 11.10.2.1, page 372).

The table lists the revisions of the object and, apart from the generally available information (see section 11.10.2.2, page 372) 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's metadata (e.g. definition of access rights via the metadata).

Selection:

For information on restoring a revision, see section 11.10.2.3, page 374. For details of comparing two revisions of a page reference, see section 11.10.2.5 page 377.

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 changes to media:** If this checkbox is activated, changes to media that are referenced to this page are displayed.

**Show changes to templates:** If this checkbox is activated, changes to templates that are used for this page are 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 partially 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 gray.




*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 datasets (both the number and the editorial content) do not correspond to the content of the datasets at the time of the revision. This behavior applies to all dependencies within an object, in other words also page references and the dependent pages, for example. The relevant content should therefore be explicitly checked after being restored.*

## 11.11 Show dependencies via reference graph

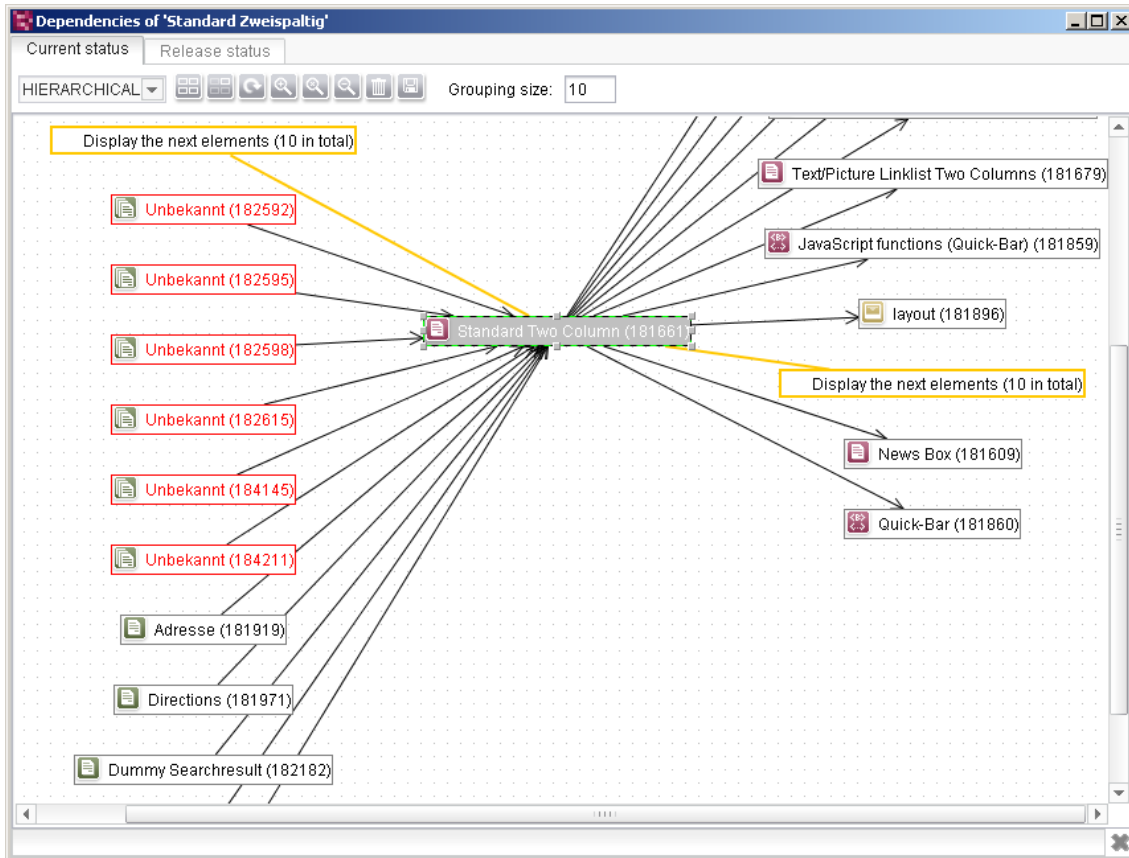
Essential functions of FirstSpirit are based on the so-called reference graph of a project. This had to be calculated for a project for the first time ever in Version 3.1 and since then has been constantly extended and further developed. The reference graph of a project is used to recognize dependencies within the project and is therefore an essential component of complex functions, for example, the server-side release.

Reference graphs (see Figure 11-91) can be requested for an object via the **Extras/Display dependencies** context menu or the keyboard shortcut <Ctrl>+<R>. Reference graphs of individual datasets of the Content Store are called via the context menu of the respective dataset.



 *This function is available to project administrators only.*

The tabs 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 11-91: 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



11-91).

**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 image: this icon can be used to save the view on the workstation computer in the .png format for use at a later date.

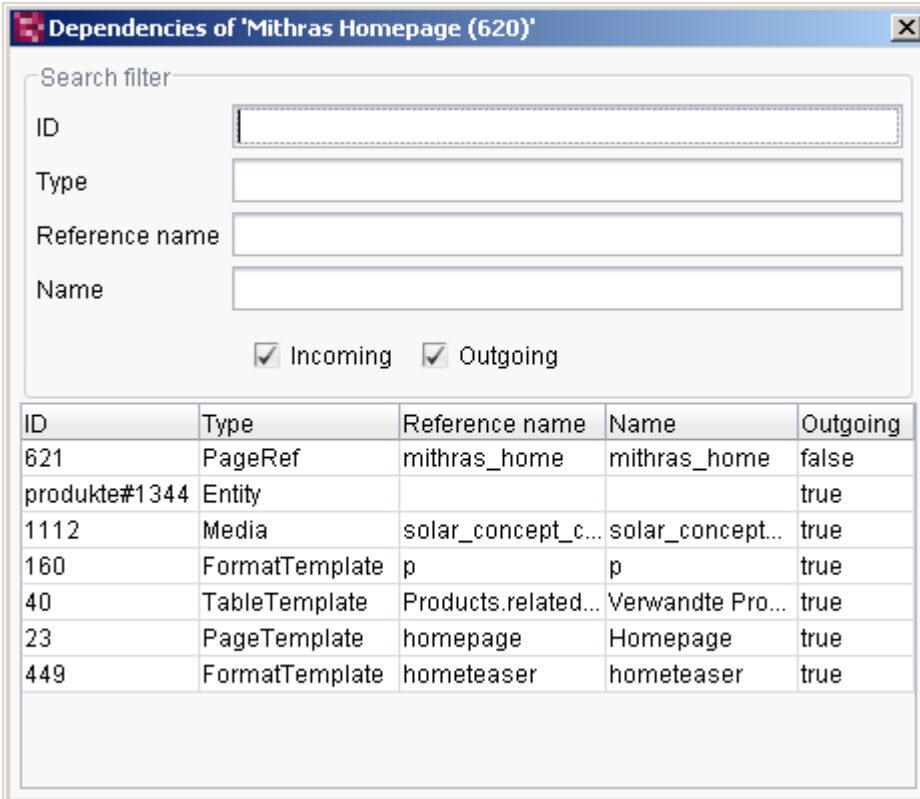
**Grouping size:** As a default, when the reference graph is opened, only the first 10 dependent elements are displayed as a maximum. This input field can be used to change the maximum number. A re-defined grouping size does not have an effect until the next time the "Display the next elements" function 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:





ID	Type	Reference name	Name	Outgoing
621	PageRef	mithras_home	mithras_home	false
produkte#1344	Entity			true
1112	Media	solar_concept_c...	solar_concept...	true
160	FormatTemplate	p	p	true
40	TableTemplate	Products.related...	Verwandte Pro...	true
23	PageTemplate	homepage	Homepage	true
449	FormatTemplate	hometeaser	hometeaser	true

**Figure 11-92: 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.



*It is also possible to use the reference graphs to display format templates that are used within the Rich text editor and Tables input forms.*



## 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 authorized groups of people can be specified for the tasks to be executed between these two states.

The authorization or permission to start a workflow is defined within the FirstSpirit editing environment using the "Permission assignment" dialog (see section 13.2, page 427).

The structure (task sequence) and properties (e.g. without context) of a workflow and definition of the authorized 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 section 13.2.5, page 432).

### 12.1 FirstSpirit standard workflows

In FirstSpirit there are two integrated workflows:

1. **Task** workflows for general completion of tasks within the project. Authorized 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 section 12.3 page 393.



*Workflows are created and configured for a project by the template developer. Both the possible actions during the processing of a workflow and the display in the FirstSpirit JavaClient (e.g. the color coding) can greatly differ from the standard workflows described here.*



## 12.2 Starting a workflow

FirstSpirit recognizes two types of workflows:

**Workflows without context** 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 workflow without context. (see section 12.2.2 page 393).

**Context-bound workflows** are tied to a specific object within the project and are always started and executed with reference to this object (see section 12.2.1, page 391). The standard "Request release" workflow is an example of a context-bound workflow (see section 12.3, page 393).



*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 key 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 12-3 and Figure 12-5, see section 12.3.1, page 394).*

### 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 (see section 4.1.11, page 134).



**Figure 12-1: Start context-bound workflow**

All context-bound workflows from the project's Template Store are listed under this context 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 section 13.2, page 427).*







*Depending on the template developer's settings, workflows can also be started by means of 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 12-3). 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 section 12.3.1 page 394).

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", the menu now shows the "Workflow Action" entry:



**Figure 12-2: Forward context-bound workflow**

All actions that have to be executed to switch the active workflow to the next state are listed under this context 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 section 13.2.5, page 432).*



*Depending on the template developer's settings, workflows can be forwarded by means of 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 forwarded via the task list (see section 12.3.2, page 398).



### 12.2.2 Workflows without context

A workflow without context can be started using the "Start Workflow (without context)" function in the "Tasks" menu of the FirstSpirit menu bar (see section 3.1.2.2, page 40).

Analogous to the execution of a context-bound workflow, 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 section 12.3.1 page 394).

Each workflow without context can be started any number of times within a project. If a workflow without context has been started the workflow can be passed forward via the task list (see section 11.1 page 305).

## 12.3 Release of objects (standard sequence)

In all FirstSpirit projects that carry out a release check, objects (such as 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 (see introduction to section 12 onward). 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 section 13.1.4.9, page 424).

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 section 12.3.1)
- Request a release via the task list (see section 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 section 12.2.1, page 391). The editor opens the standard workflow using the "Start workflow" context menu entry, sub-menu entry "Request release".

The "Workflow Action" dialog box opens for starting the workflow:

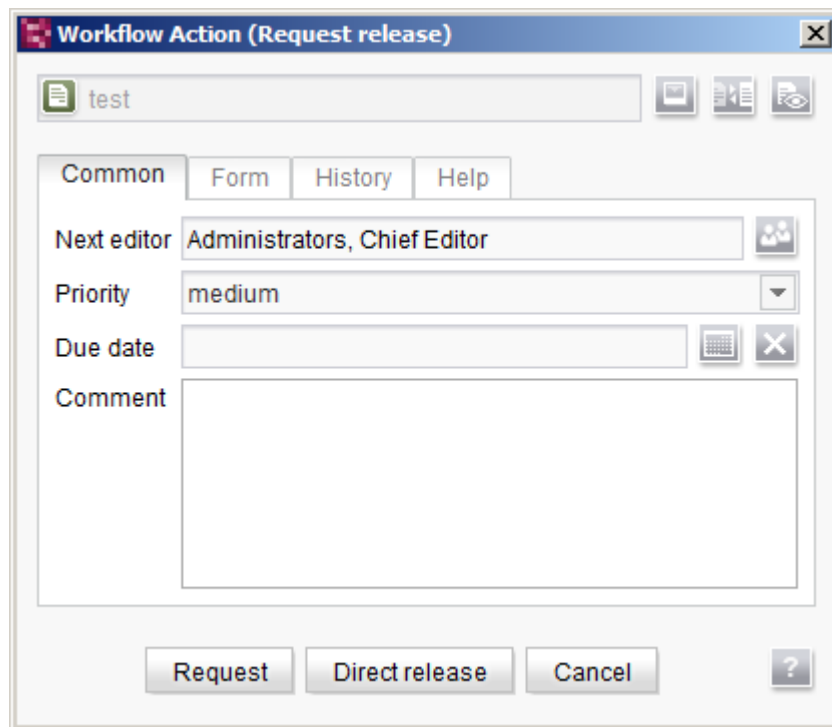




Figure 12-3: Workflow action (request release) dialog box


Various settings can be made or read in this action window.


#### General tab

The top field displays the object on which the workflow was opened. After this row, there are three icons for checking the object:



-  Go to object: click this icon to display the object in the tree structure.
-  Compare: click this icon to open a version comparison for the respective object (see also section 11.10.2.5, page 377). 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.



-  Display preview: click this icon to display the preview of the object to be released. In this way changes can be simply checked before release and if necessary can be revised.

**Next editor:** The required editor for this workflow can be selected using the  icon after the row (for information on the assignment of users or groups, see section 13.2.4, page 431).

**Priority:** This can be used to set the priority to be given to this workflow during the editing or processing.

**Due date:** Click the  icon to specify a deadline by which the whole workflow must have been completed. Click the  icon to remove the deadline.


**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.

A button for each state that can be achieved is displayed in the lower part of the workflow dialog. 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 this 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.

**Direct release:** Click this button to release the object directly and end the workflow. The labeling 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 box; a release is not requested.

The editor can use the  icon or "Help" 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):



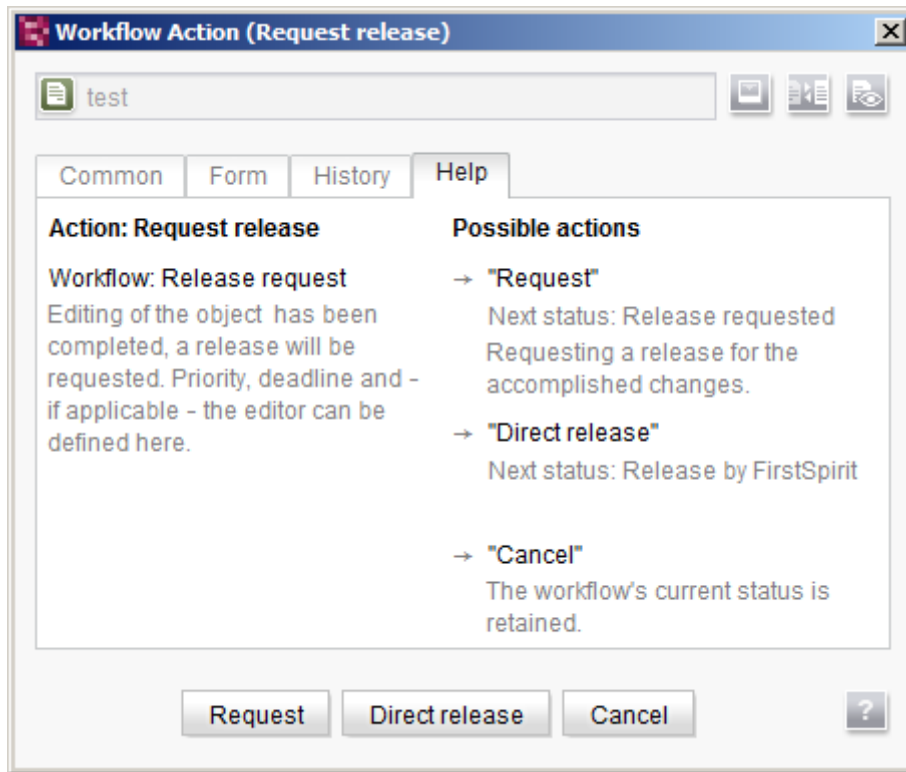


Figure 12-4: Workflow action – "Help" tab

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, tool tips, elements of a combo box, etc. can be displayed language-dependent.

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

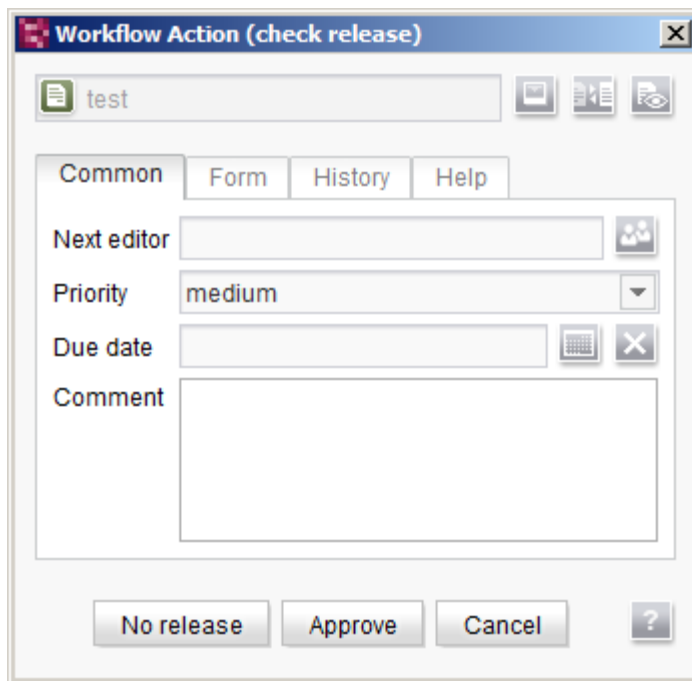


*If the workflow has been started on several objects only the options "Priority", "Due date", "Comment" and the actions are available.*

If a release has been requested the object is passed forward to the **next status** of the workflow. As a result, the color of the label in the tree structure changes and the context menu also changes (see section 12.2.1, page 391). The started "Request release" workflow can be forwarded. Click the context menu "check" entry to open the "Workflow action" dialog box again.

New buttons are now available in the "Actions" area of the window only.





**Figure 12-5: Workflow action**

**No release:** If the button is clicked, the release is rejected 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.

**Approve:** 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 box; 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.*



### 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  icon in the tool bar or using the "Tasks – Task list" menu item of the menu bar.

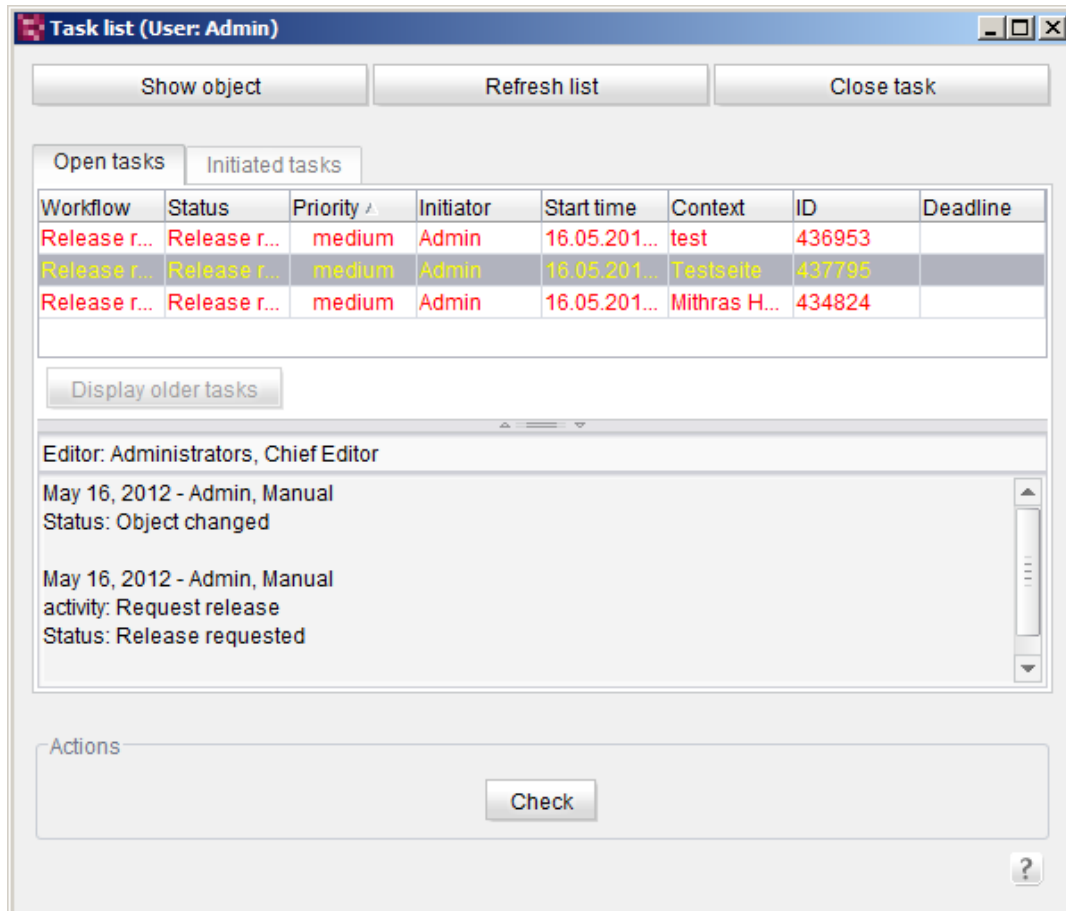


Figure 12-6: Task list

The layout and functions of the task list are documented in detail in section 11.1, page 305.

**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:

<u>Status</u>	<u>Possible actions (buttons)</u>
Object changed	request release
Release requested	check (release/do not release)
Object not released	edit
Object released	final status



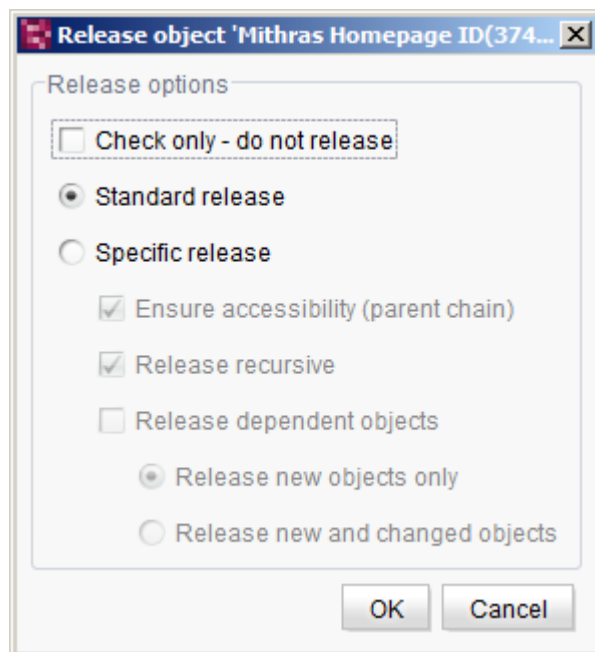
If a release has been requested for an object from the Open Tasks list the object is switched to the next workflow status. As a result, the color of the label in the tree structure changes and the button in the "Actions" window area also changes. Click the **check** button to open the "Workflow action" dialog box again. The following release steps are analogous to the release of an object using the context menu (see section 12.3.1, page 394).

## 12.4 Special release options



*The special release options are only available to project administrators. These special release options are not available to editors with the "Release" permission.*

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 12-7: Release options for direct release**

**Comment:** Similar to the assignment of comments when leaving editing mode (see section 3.2.5, page 78), 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, image 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 grayed 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 that were previously never released are also released, starting from the selected object. 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 section 12.4.1 page 402).

**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 section 12.4.2 page 403)

**Specific release – Release dependent objects:** If this option is selected all objects dependent on the selected object (e.g. a medium used in an image input component) are released also. A differentiation is made between objects that have never been released to date (**Release new objects only**) and objects that have been re-edited after they had been released (**Release new**



and changed objects) (for an example, see section 12.4.3, page 404).

 *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. Conversely, 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 **Display details** 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.

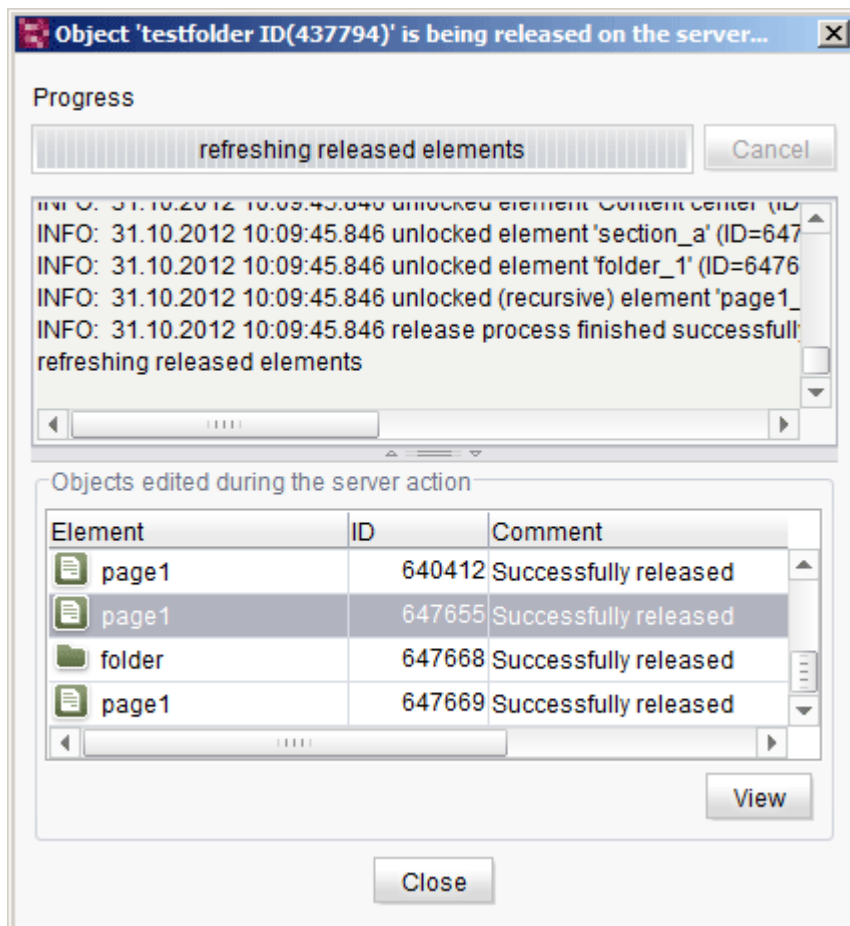


Figure 12-8: Detail window for special release

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.

**Display:** Click this button to display the objects selected from the list in the JavaClient's tree structure.

### 12.4.1 Example: Specific release - Ensure accessibility (parent chain) option

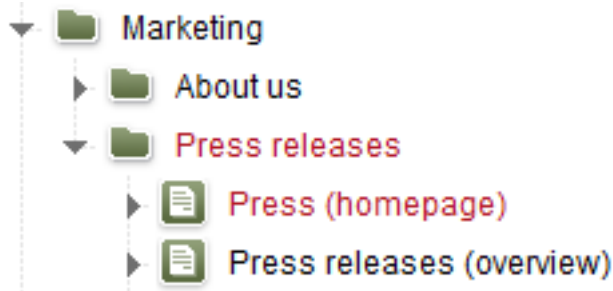


Figure 12-9: Initial situation for Release parent chain

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.

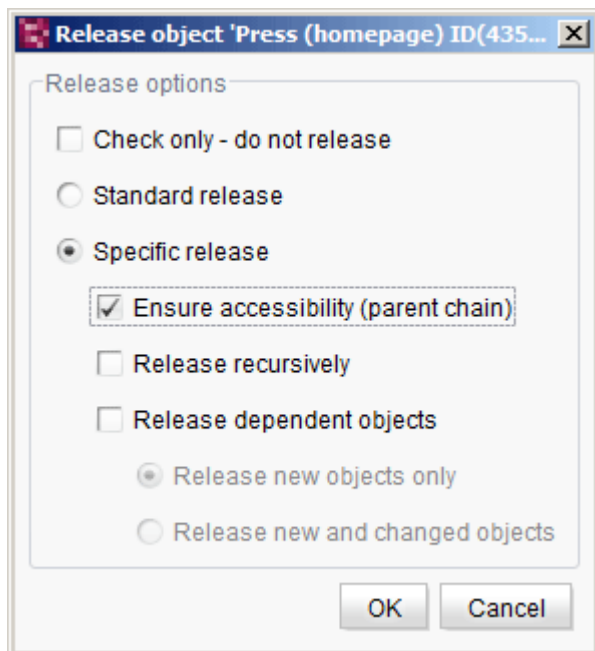
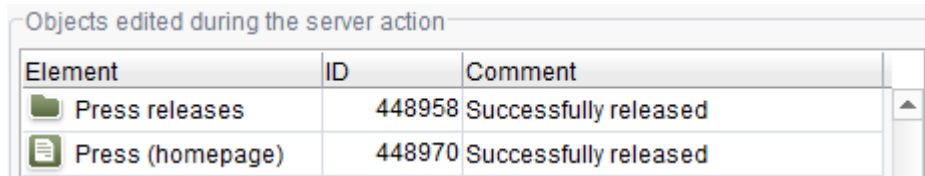


Figure 12-10: "Release parent chain" option



Result of the release:



Element	ID	Comment
Press releases	448958	Successfully released
Press (homepage)	448970	Successfully released

Figure 12-11: Result of the release

#### 12.4.2 Example: Specific release – Release recursively

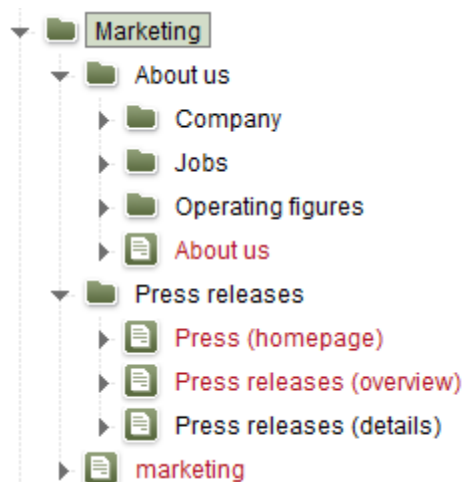


Figure 12-12: Initial situation for Release recursively

If the "Specific release" is now executed for the higher-level folder with the "Release recursively" option, all lower-level pages of the folder are released also.



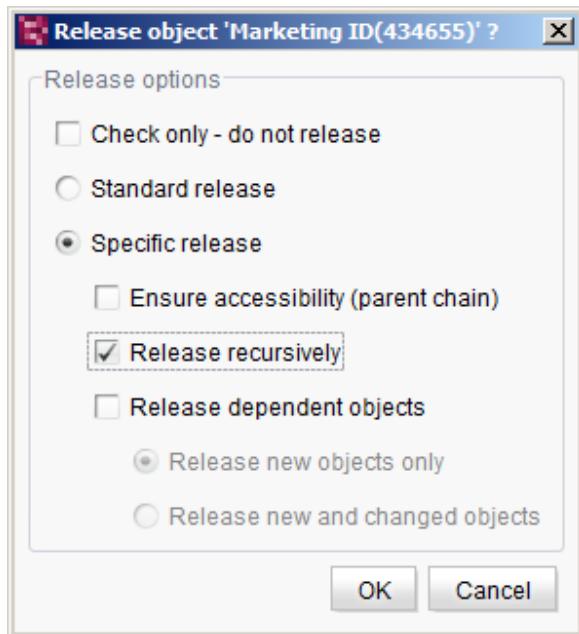


Figure 12-13: "Release recursively" option

Result of the release:

Objects edited during the server action		
Element	ID	Comment
marketing	448978	Successfully released
About us	435471	Successfully released
Press releases (ove...	434980	Successfully released
Press (homepage)	435832	Successfully released

Figure 12-14: Result of the release

### 12.4.3 Example: Specific release – Release dependent objects

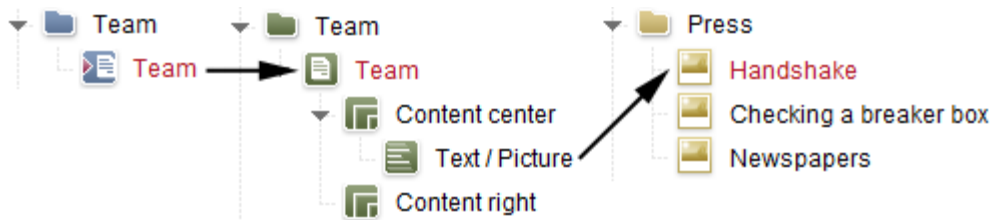


Figure 12-15: Initial situation for Release dependent objects

If a medium is changed (or re-uploaded) via the image 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 that this page references is not released, as from the view of the page (Page Store) it is not an outgoing reference.

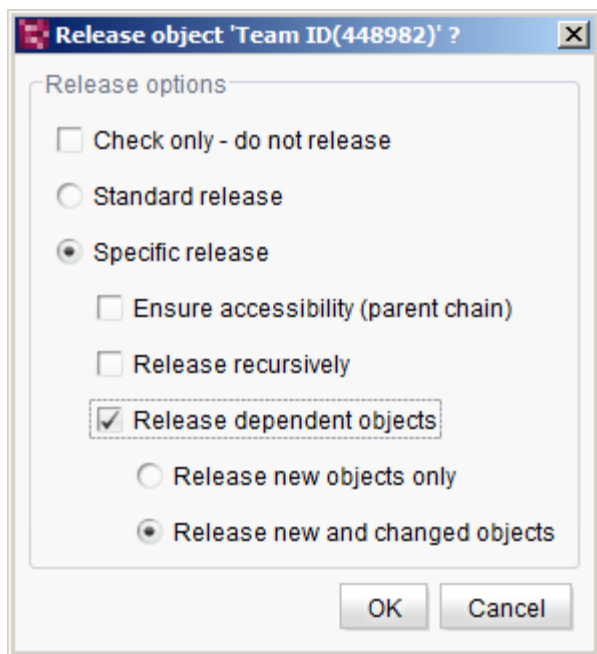


Figure 12-16: Option "Release dependent objects"



According to the settings in the project configuration, this dialog enables you to assign a comment, too.

Result of the release:



Objects edited during the server action		
Element	ID	Comment
 Team	448982	Successfully released
 Handshake	435353	Successfully released

Figure 12-17: 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 section 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 the permissions that apply for a user of FirstSpirit. These permissions are initially issued by assigning the user to a project group and can be further specified by authorized persons within the FirstSpirit JavaClient (see section 13.1, page 407).
- **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 (see section 13.2, page 427).  
The Developer's Manual contains documentation about assigning these permissions (section "4.7 – Permissions system for workflows").
- **User permissions:** Permissions valid for the "visitor" to the site generated with FirstSpirit. User permissions are always linked with the personalization system used (see section 13.3, page 434).

### 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 section 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 section 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

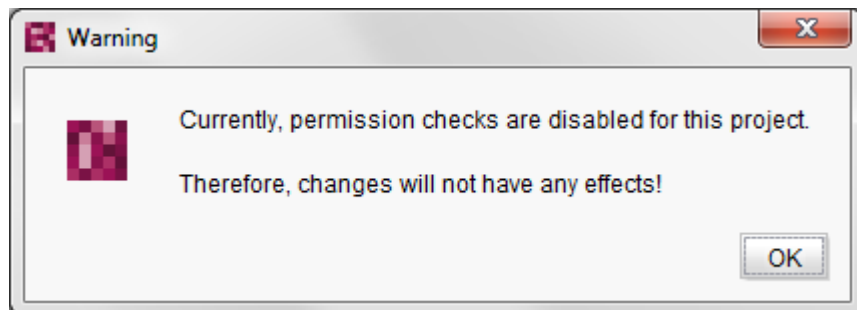




user to the project using the administrator console. Advanced editorial permissions can now either be granted directly to the project user or they can be allocated to the project user through the assignment to a project group with the corresponding permissions (see section 13.1.1).

The permissions to be assigned are explained in greater detail in section 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 by default all permissions in JavaClient since FirstSpirit version 5.0R2 (before he had limited 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 permission 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 permission assignment for a group in the FirstSpirit JavaClient (see section 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 section 13.1.3, page 416).

The "Extras/Change permissions" function is now called on the selected object via the content menu. The permission assignment dialog opens.

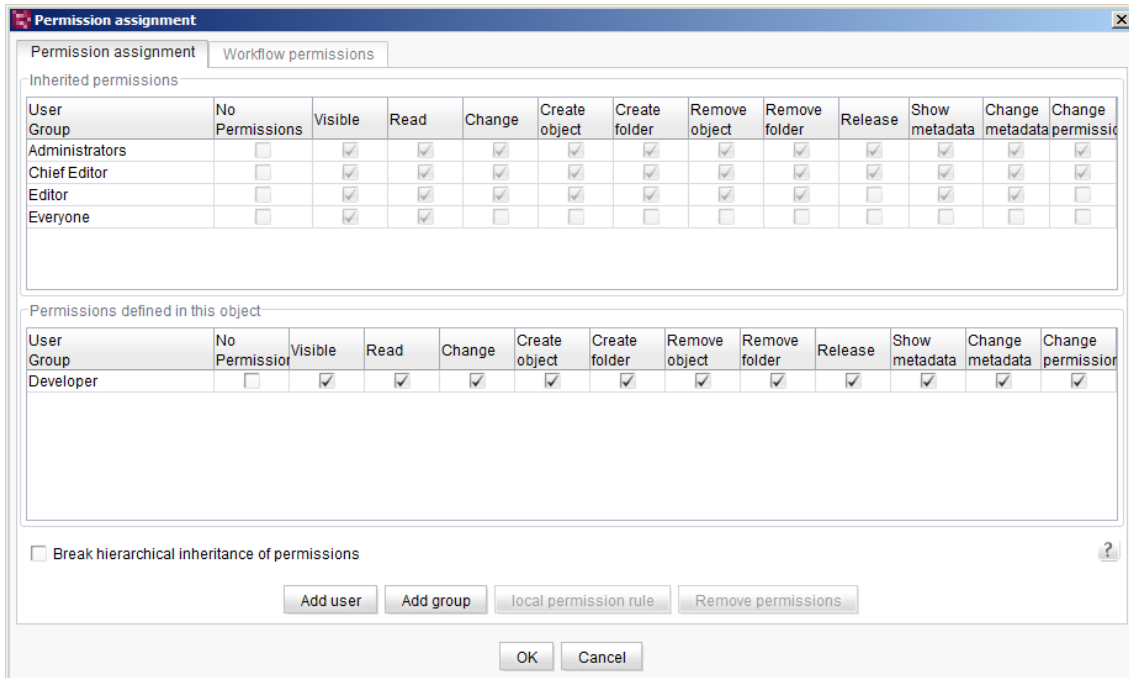



Figure 13-2: Permission assignment dialog – "Permission assignment" tab


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 section 13.2 page 427.

By right-clicking  you can choose whether to display the login, name or name and first name of the users.

 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 (except on the root node of the stores). The bottom area "Permissions defined in this object" is initially empty as permissions have not yet been defined for this node.

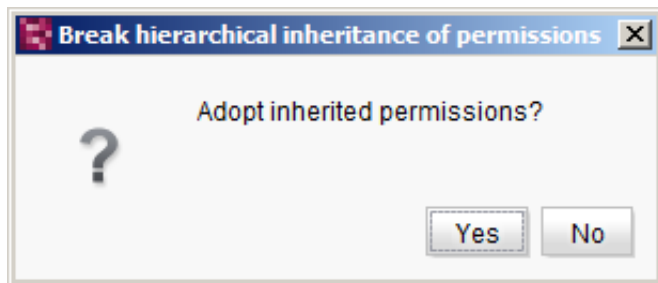


 For more clearness the entries in the lists "Inherited permissions" and "Permissions defined in this object" are sorted automatically in alphabetical order. First groups and then users are shown.

If, in addition to the inherited permissions, further permissions are defined for a group or a user, the corresponding group (see section 13.1.2.3, page 412) or user (see section 13.1.2.4, page 415) must be added to the 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 section 13.1.2.1, page 411).

### 13.1.2.1 Adopt inherited permissions for a node



**Figure 13-3: Adopt inherited permissions**

**No:** If the dialog is confirmed with "No" the permission 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.

Permissions defined in this object			
User Group	No Permissions	Visible	Read
Everyone	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Figure 13-4: 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:



Permissions defined in this object			
User Group	No Permissions	Visible	Read
Everyone	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Chefred...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Redakte...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Marketing	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

**Figure 13-5: Permission 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 section 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 section 13.1.2.6 page 416).

### 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 "Permission assignment" dialog are saved if the changed permissions are confirmed by clicking "OK".

**Cancel:** Click to cancel the assignment of permissions. Permission definitions that have already been changed or deleted 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.

**Add groups:** Click this button to open the "Add groups" dialog with a list of all project groups:



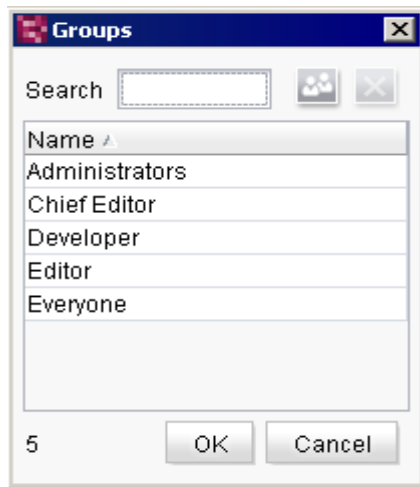



Figure 13-6: Add group

A project group can be selected from the list. Click the  icon to add the selected group to the "Permission assignment" dialog where it is included in the "Defined permissions in this object" area.

Permissions defined in this object													
User Group	No Permissions	Visible	Read	Change	Create object	Create folder	Remove object	Remove folder	Release	Show metadata	Change metadata	Change permissions	
Everyone	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Chefred...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Redakte...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Marketing	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Figure 13-7: New group added

The permissions for the new group or user added can now be defined (see section 13.1.2.2, page 412).



*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 this group but the inherited permissions are not deactivated (see section 13.1.2.1, page 411), 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.



Example for adding the Everyone group

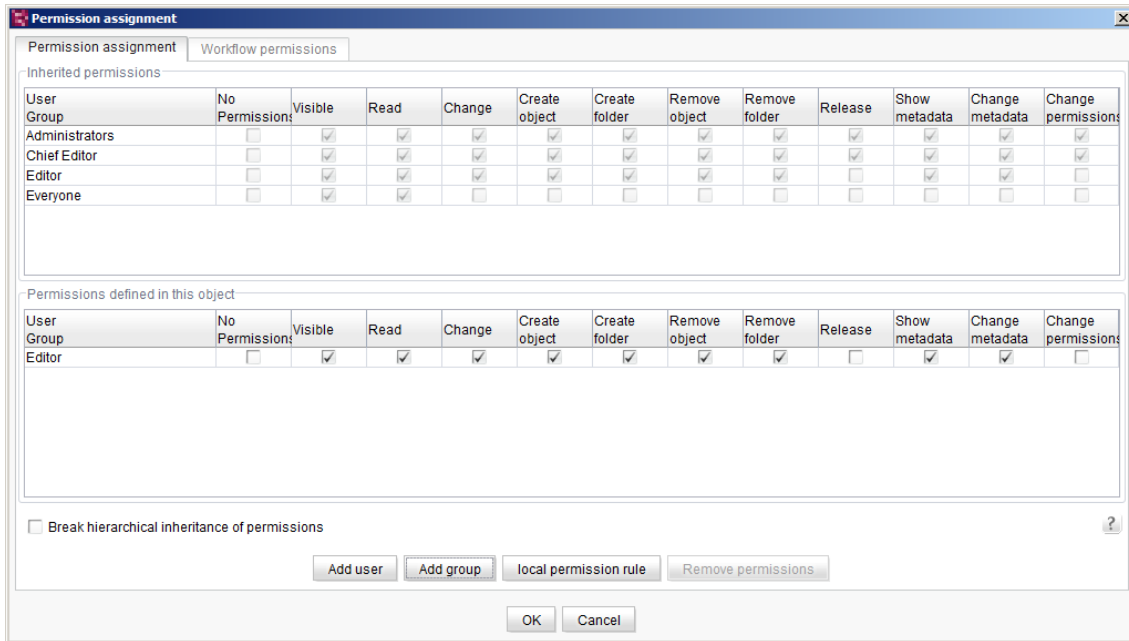


Figure 13-8: Add "Everyone" group – Initial configuration

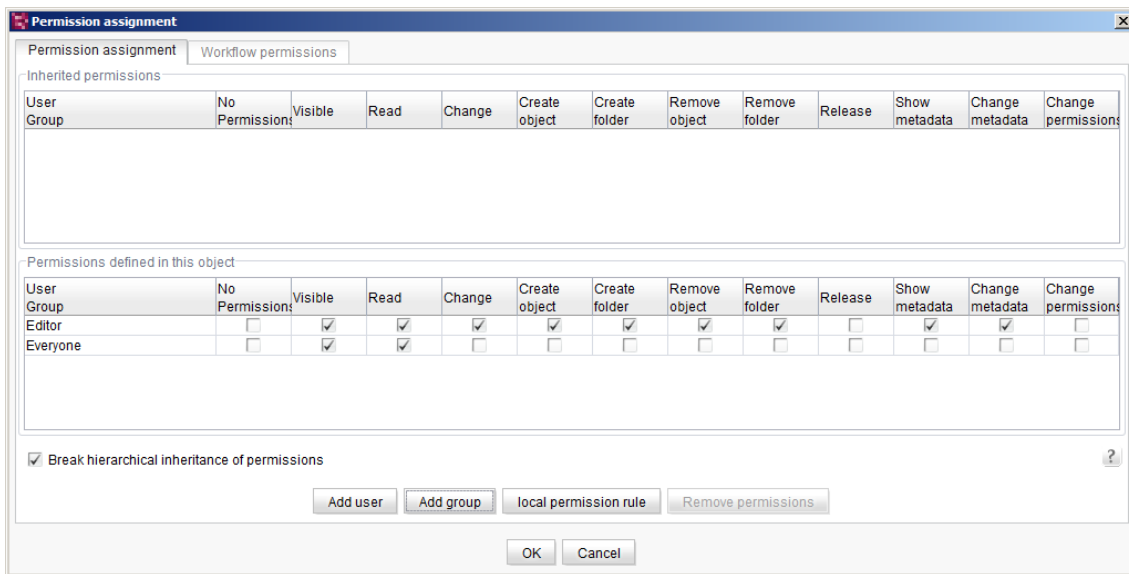


Figure 13-9: Add "Everyone" group – Result

If other groups are selected at the same time as "Everyone", the permissions of these groups are also adopted.



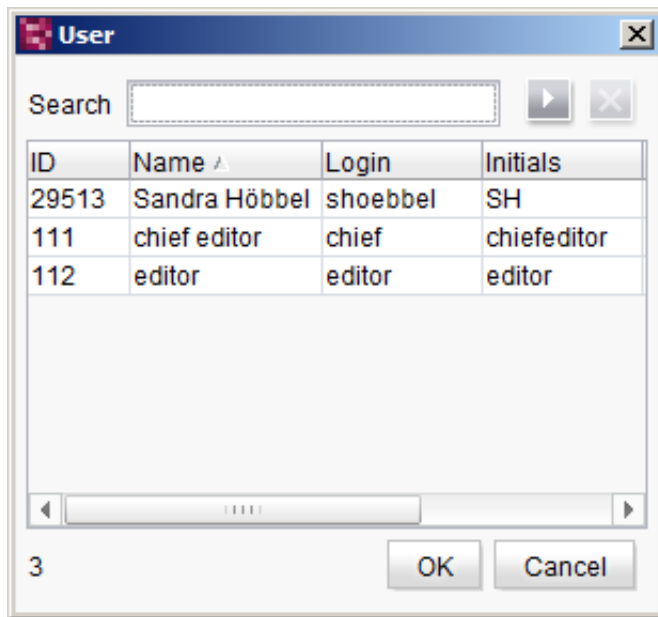


When this action is carried out, a confirmation prompt appears informing you that the hierarchical inheritance of permissions will be broken. If it is confirmed with "Yes", the response is as described above, if "No" or "Cancel" is selected, all selected groups except for "Everyone" are added. Therefore, 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.

**Add user:** Click this button to open the "Add user" dialog with a list of all project users:



**Figure 13-10: Add user**

A project user can be selected from the list. Click **OK** to add the selected user to the lower part of "Permission assignment" where it is included in the "Defined permissions in this object".

Permissions can now be defined for the new user added (see section 13.1.2.2, page 412).







*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 section 13.1.2.1, page 411) 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

**As local permissions 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 permissions rule" is used, the inheritance still has to be noted and observed (see section 13.1.2.1, page 411).*


### 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. This applies to the Page Store, Site Store, Media Store and the global pages in the Global Settings. 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 section 13.1.2.1, page 411). 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 on which permissions were explicitly assigned are indicated by the  icon. However, the permissions symbol only appears in the tree view if the "Show symbols" setting was activated in the "View" menu (see section 3.1.5.1, page 55).

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.

*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

Visible	Read	Change	Create object	Create folder	Remove object	Remove folder	Release	Show metadata	Change metadata	Change permissions
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

**Figure 13-11: 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 gray) 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 section 13.1.5, page 425).

#### 13.1.4.3 The "Visible" and "Read" permissions

FirstSpirit's security model differentiates between the **editorial content** and the **internal project information** of a FirstSpirit 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** that 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.

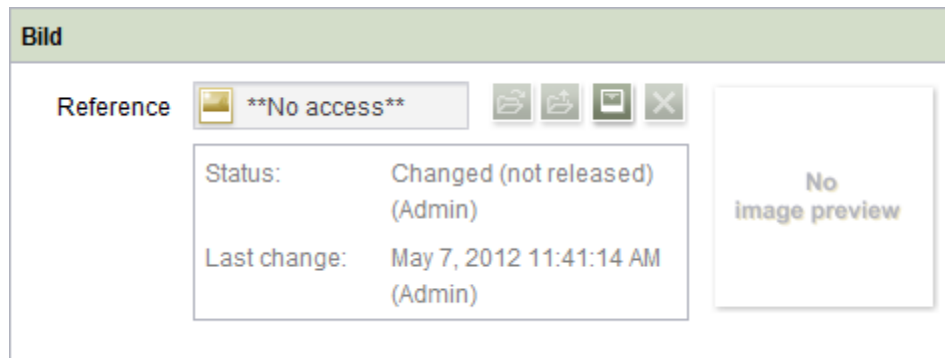
**Read:** The "Read" permission is an access protection that prevents access to the data or content of the object protected in this way. Therefore, access to objects via the FirstSpirit Access API which are protected by the "Read" permission ("Read" permission is withdrawn), immediately generates a security 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 Permission	Visible	Read
Chief Editor	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

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 13-12: "Visible" permission has been withdrawn for the medium**

User Group	No Permissions	Visible	Read
chief (chief editor)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

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:



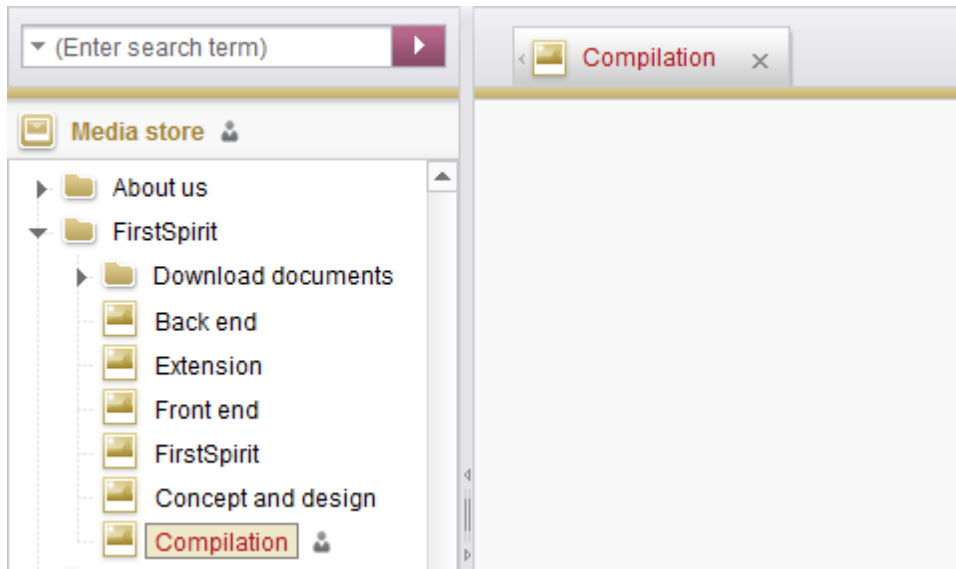


Figure 13-13: "Visible" permission has been set for the medium

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"):

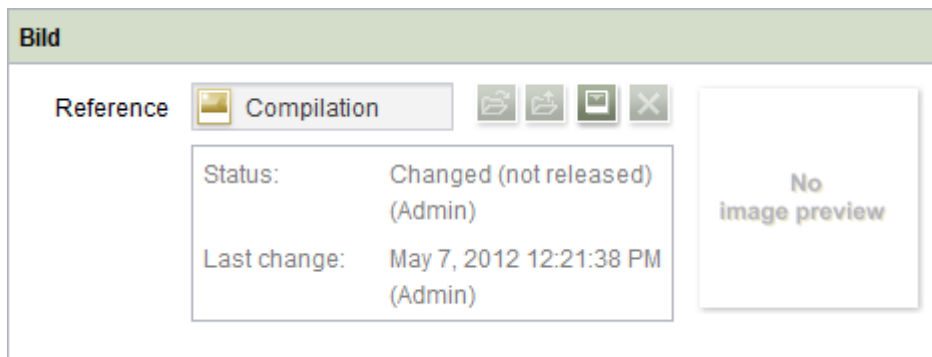


Figure 13-14: "Visible" permission has been set for the medium – Display in the input component

User Group	No Permissions	Visible	Read
chief (chief editor)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

If the **Visible** permission and the **Read** permission have been set for an object, this is displayed both in the project structure (tree view) and with its editorial content.



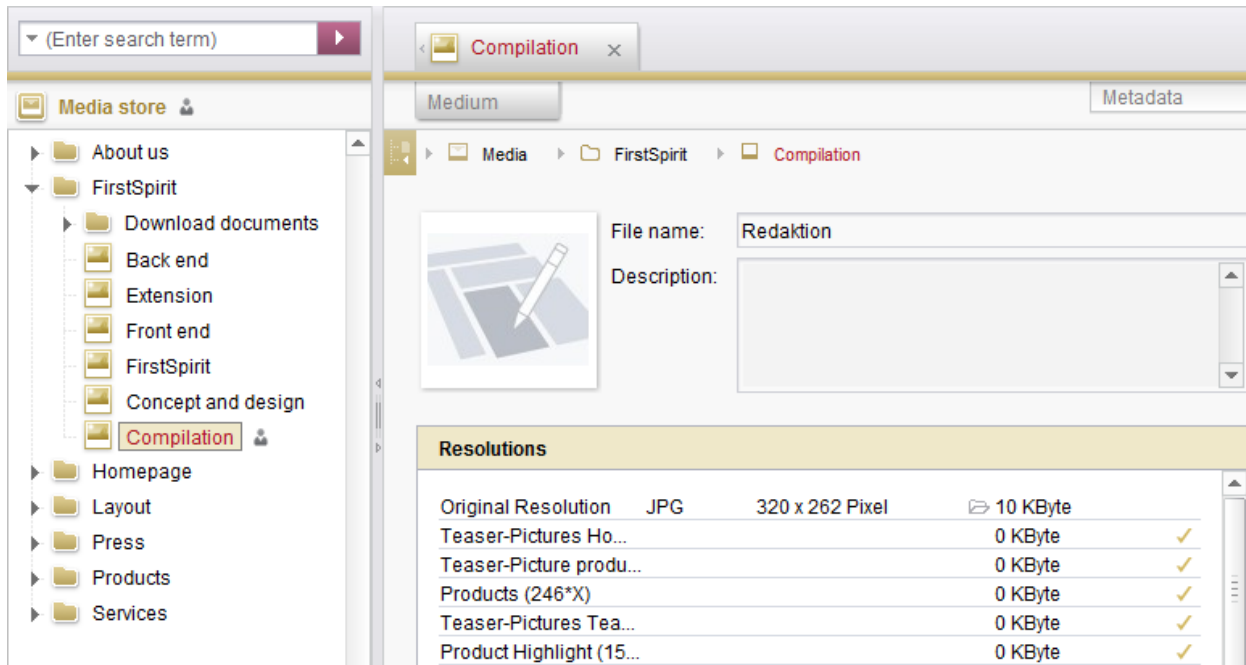


Figure 13-15: The "Visible" and "Read" permissions have been set for the medium

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, for example in the form of an image preview:

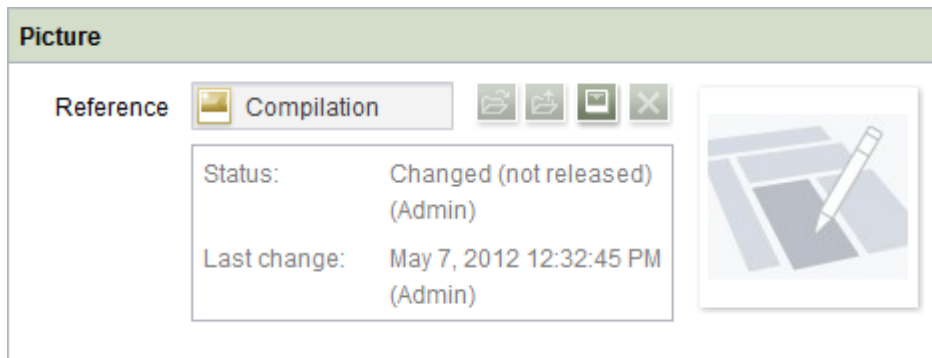


Figure 13-16: The "Visible" and "Read" permissions have been set for the medium



#### 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 editing mode (see section 3.2.5, page 78).
- Change contents of the object.
- In the Page Store this permission also relates to:
  - Deleting sections
  - Adding sections
  - Copying sections

The "Change" permission has dependences on other editorial permissions (see section 13.1.5, page 425).

#### 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 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.*





*At the same time, the user needs 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 section 13.1.5, page 425).

#### 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 and filters:
- 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 section 13.1.5, page 425).

#### 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 section 13.1.5, page 425).







*Deleted objects can be restored using the "Delete" icon (see section 3.2.8, page 82). 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 section 13.1.4.6, page 423).

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. Of course, this also applies if the user has the "Delete" permission for all lower-level objects.

If there are elements below a folder for which the user does not have permission to delete, these elements are retained (together with the folder as a parent node).

The "Delete folder" permission has dependences on other editorial permissions (see section 13.1.5, page 425).

#### 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 section 12.3, page 393).

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 assigning workflow permissions (see section 13.2, page 427).

The "Release" permission has dependences on other editorial permissions (see section 13.1.5, page 425).



#### 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 content and is different in a project-specific way (see section 11.3, page 309). A special form of metadata is, for example, user permissions, whose maintenance via metadata is explained in section 13.3.2 (page 435).

If the "Show metadata" permission is granted, metadata that has already been entered is displayed to the user (for example, already assigned user permissions).

The "Show metadata" permission has dependences on other editorial permissions (see section 13.1.5, page 425).

#### 13.1.4.11 Permission: Change metadata

If the "Change metadata" permission is granted the user can make changes to the content of the metadata.

The "Change metadata" permission has dependences on other editorial permissions (see section 13.1.5, page 425).

#### 13.1.4.12 Permission: Change permissions

If the "Change permissions" permission is granted, the user can execute the permission assignment for groups and users described in this section. It is advisable to only grant this permission to persons who are assigned the role of a project administrator. By default (in newly created projects), project administrators have got this permission.

The "Change permissions" permission has dependences on other editorial permissions (see section 13.1.5, page 425).

### 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: <ul style="list-style-type: none"> <li>▪ "Visible"</li> <li>▪ "Read"</li> </ul>
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: <ul style="list-style-type: none"> <li>▪ "Visible"</li> <li>▪ "Read"</li> </ul>
Show metadata:	Dependence on the permissions: <ul style="list-style-type: none"> <li>▪ "Visible"</li> <li>▪ "Read"</li> </ul>
Change metadata:	Dependence on the permissions: <ul style="list-style-type: none"> <li>▪ "Visible"</li> <li>▪ "Read"</li> <li>▪ "Change"</li> </ul>
Change permissions:	Dependence on the permissions: <ul style="list-style-type: none"> <li>▪ "Visible"</li> <li>▪ "Read"</li> <li>▪ "Change"</li> </ul>



## 13.2 Permissions to execute workflows

Workflow permissions are a special type of editorial permission that only relate to the workflows within a project (for further information on workflows, see section 12, page 390). Workflow permissions are assigned parallel to the editorial permissions for groups and users within the stores in the FirstSpirit JavaClient (see section 13.1.1, page 409).

The permissions are assigned in the same ways as the assignment of user permissions using the "Permission assignment" dialog (see section 13.1.2, page 409).

### 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.

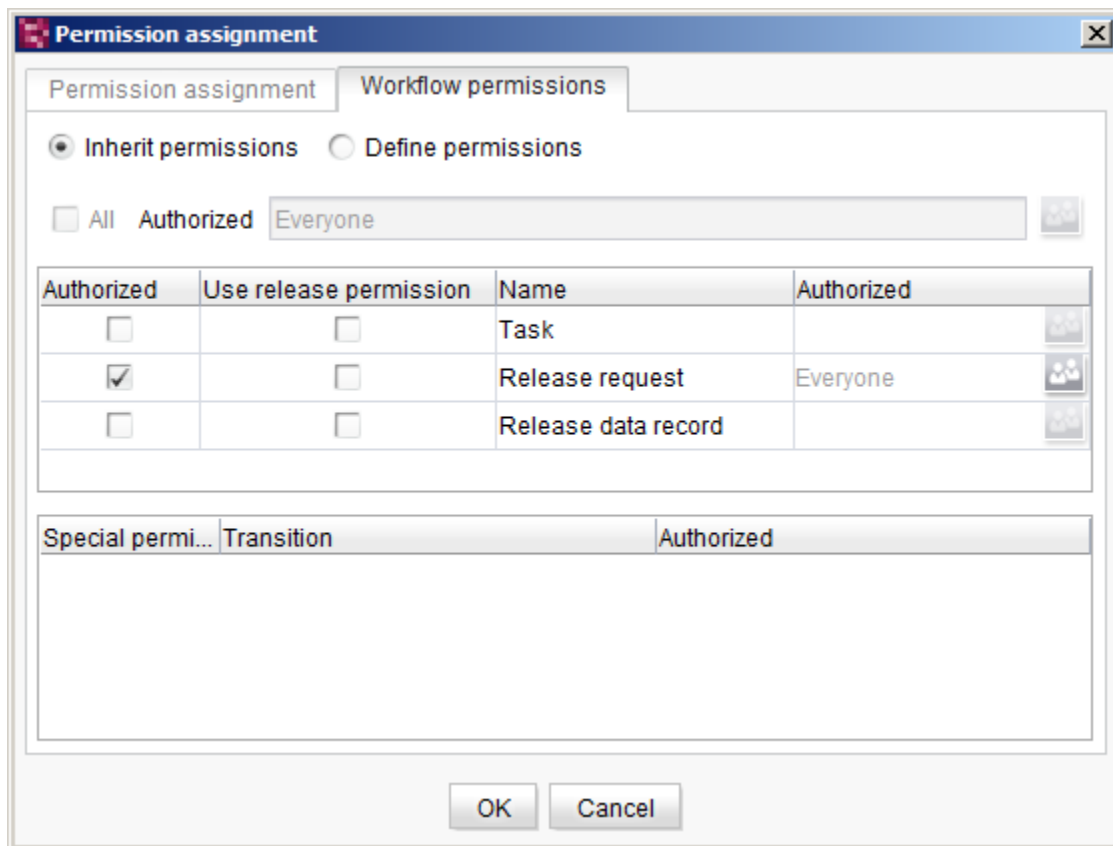


Figure 13-17: Permission assignment dialog - "Workflow permissions" tab



Users authorized 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" radio button is selected as a default (exception: root nodes). With this setting, the "Workflow permissions" are inherited from a high-level node (see section 13.1.3, page 416).

**Define permissions:** The "Define permissions" radio button must be activated if the permission definition for the execution of a workflow on the current node is to be changed. Then a question appears asking whether or not you want to adopt the inherited permissions (see section 13.1.2.1, page 411). Permissions defined on a node are inherited by all lower-level objects (see section 13.1.3, page 416).

If the dialog is confirmed with **No**, the permissions definition for this node starts with deactivated permissions (see Figure 13-17). In this case the permissions set to date for a group or user at a higher level node are not adopted.

If the dialog is confirmed with **Yes**, the inherited permissions are adopted as a default setting:

Authorized	Use release permission	Name	Authorized
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Task	Everyone
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Release request	Everyone
<input type="checkbox"/>	<input type="checkbox"/>	Release data record	

**Figure 13-18: Permission 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 section 13.2.2 onward).

### 13.2.2 Define permissions for all workflows

To make the permission 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.



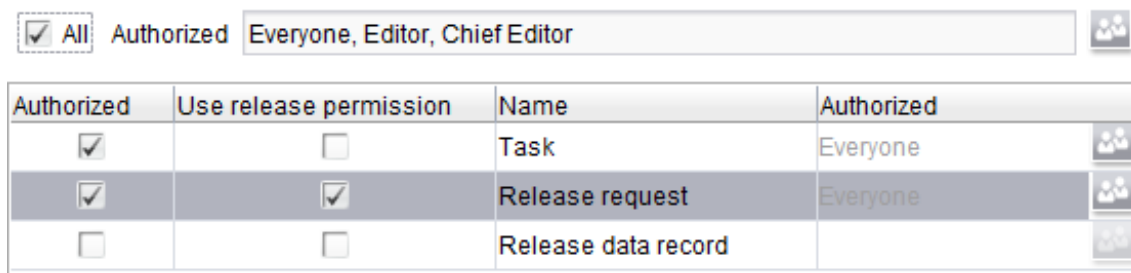



Figure 13-19: Define permissions for all workflows

**Authorized:** All users and/or groups who are authorized to execute a workflow on the current node are listed in this field (add or delete authorized persons, see section 13.2.4).



*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 section 13.2.5, page 432).*

Click **OK** to save all the changed permissions in the "Workflow permissions" dialog.

Click **Cancel** to cancel the assignment of permissions. Permission definitions that have already been changed or deleted 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 edited separately for each workflow.

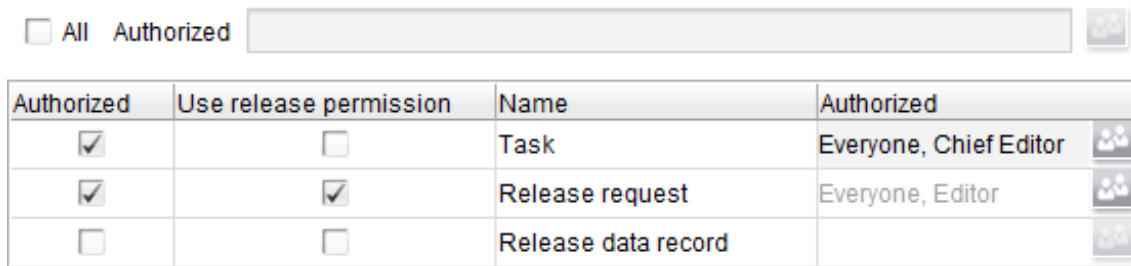


Figure 13-20: Define permissions for an individual workflow

**Authorized:** If this checkbox is activated, authorized persons may start this workflow on the current node of the tree structure. As soon as the checkbox has been activated, the permission



assignment for the current workflow can be defined in detail in the bottom part of the window (see section 13.2.5, page 432).

**Use release permissions:** If the "Use release permission" checkbox is activated the release permissions defined in the "Permission assignment" tab are evaluated for each user (see section 13.1.4.9, page 424).



*Contradictions in the permission 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 "authorized" 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 "unauthorized" 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.


**Authorized:** All users and/or groups who are authorized to execute a workflow on the current node are listed in this field (add or delete authorized persons, see section 13.2.4).

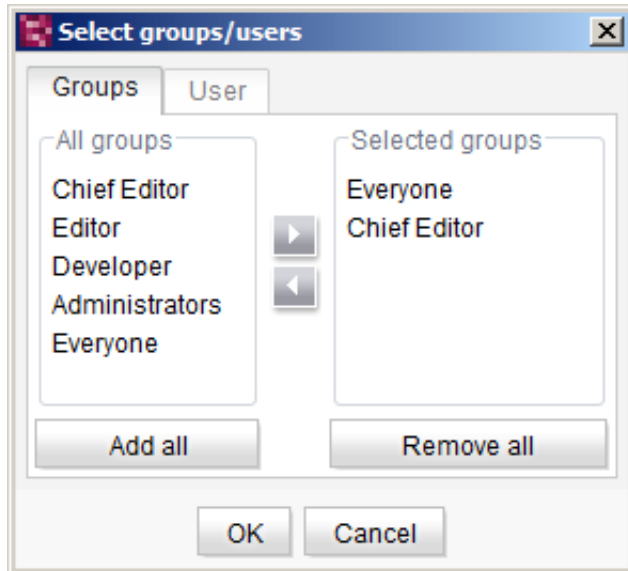
Click OK to save all the changed permissions in the "Workflow permissions" dialog.

Click Cancel to cancel the assignment of permissions. Permission definitions that have already been changed or deleted are not saved.







### 13.2.4 Change authorized groups/users

The  icon in the "Authorized" column can be used to change the selection of authorized persons. Click the icon to open the "Select groups/users" dialog.



**Figure 13-21: Select authorized 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 in a similar way 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 canceled, changes are not saved.








### 13.2.5 Permission assignment for executing the transitions

If permissions have been defined on a node for a workflow (see section 13.2.3, page 429), 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:

Authorized	Use release permission	Name	Authorized
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Task	Everyone, Chief Editor 
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Release request	Everyone, Editor 
<input type="checkbox"/>	<input type="checkbox"/>	Release data record	

Special permi...	Transition	Authorized
<input type="checkbox"/>	Objekt verändert ▶ Request release	Everyone 
<input type="checkbox"/>	Request	Everyone 
<input checked="" type="checkbox"/>	Check	Chief Editor 
<input checked="" type="checkbox"/>	No release	Chief Editor 
<input type="checkbox"/>	Edit	Everyone 
<input type="checkbox"/>	Request again	Everyone 
<input checked="" type="checkbox"/>	Approve	Chief Editor 
<input type="checkbox"/>	Direct release	Everyone 
<input type="checkbox"/>	automatische Freig. ▶ Automatic Release	
<input type="checkbox"/>	Final	Everyone 

**Figure 13-22: 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 authorized 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 authorized to execute special permissions do not have to be identical with the users or groups who are authorized to start the workflow (see section 13.2.3 page 429).*

**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.

**Authorized:** All users and/or groups who are authorized to execute this transition on this node are listed in this field (for information on adding or deleting authorized persons, see section 13.2.4).

Figure 13-22 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 (see section 12.3, page 393):

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 authorized 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 editor" 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 editor" 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 personalization system used (e.g. FirstSpirit DynamicPersonalization<sup>2</sup>). 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 section 13.1, page 407).

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 (see 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.

---

<sup>2</sup> See documentation for the module FirstSpirit DynamicPersonalization

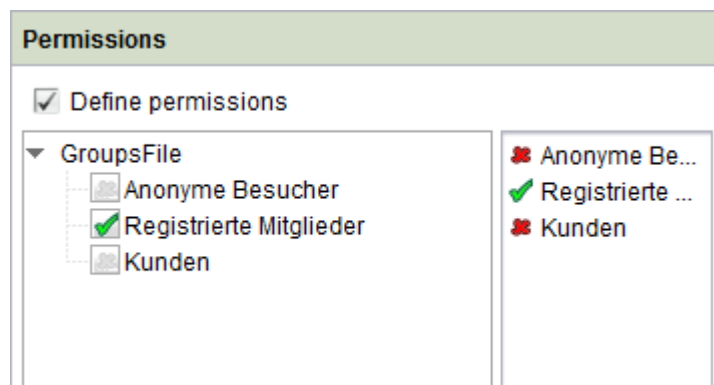


The group hierarchy is presented to the editor in the form of a tree view in which the permissions can be configured (see section 13.3.2, page 435).

### 13.3.2 Permission assignment in the FirstSpirit JavaClient

In the FirstSpirit JavaClient, the user permissions are assigned using a special input form. User permissions can be assigned on the basis of a hierarchical group definition using the permission definition form (see section 13.3.1, page 434).

The permission definition form 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 Page Store.



**Figure 13-23: Permission 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 section 13.3.4, page 437).

A tab is displayed in the input component for each user permission (for example "Visible") to be defined using the permission component. 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.

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  icon (if no other metadata has been defined) and the check mark is missing at "Define permissions".

Evaluation of the user permissions is explained in the following section 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:

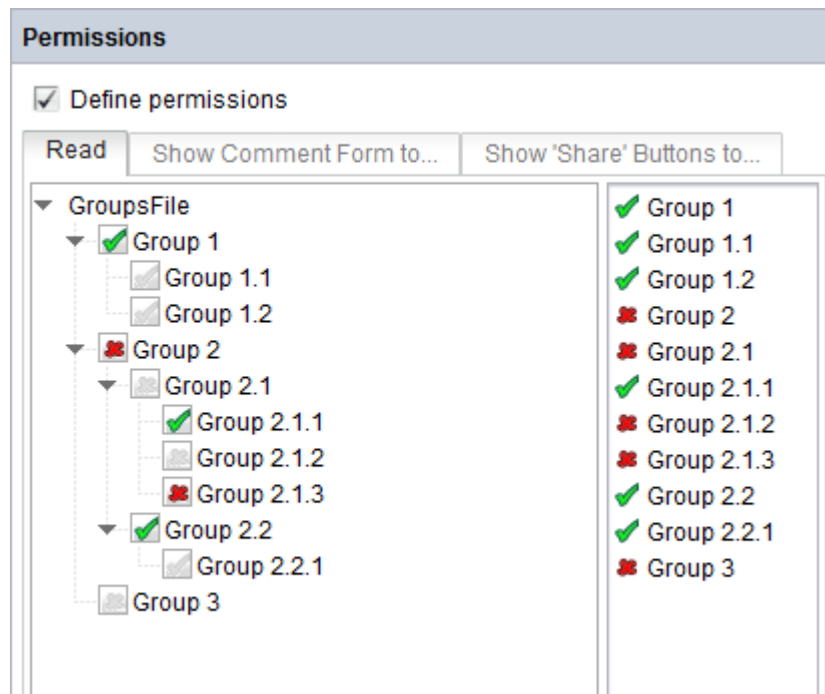






Figure 13-24: Example of permission component






Access is basically prohibited for all " doku". This means 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 authorized. 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" but not to "group 2.1.1" and "group 2.1.3"). Effectively, this configuration means that 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 authorization configuration is displayed on the right-hand side of the input component (see Figure 13-24). The calculation involves the complete group tree being 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 section 13.1.3, page 416), 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 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 authorized 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 authorization is pointless as the high-level entry point is missing.

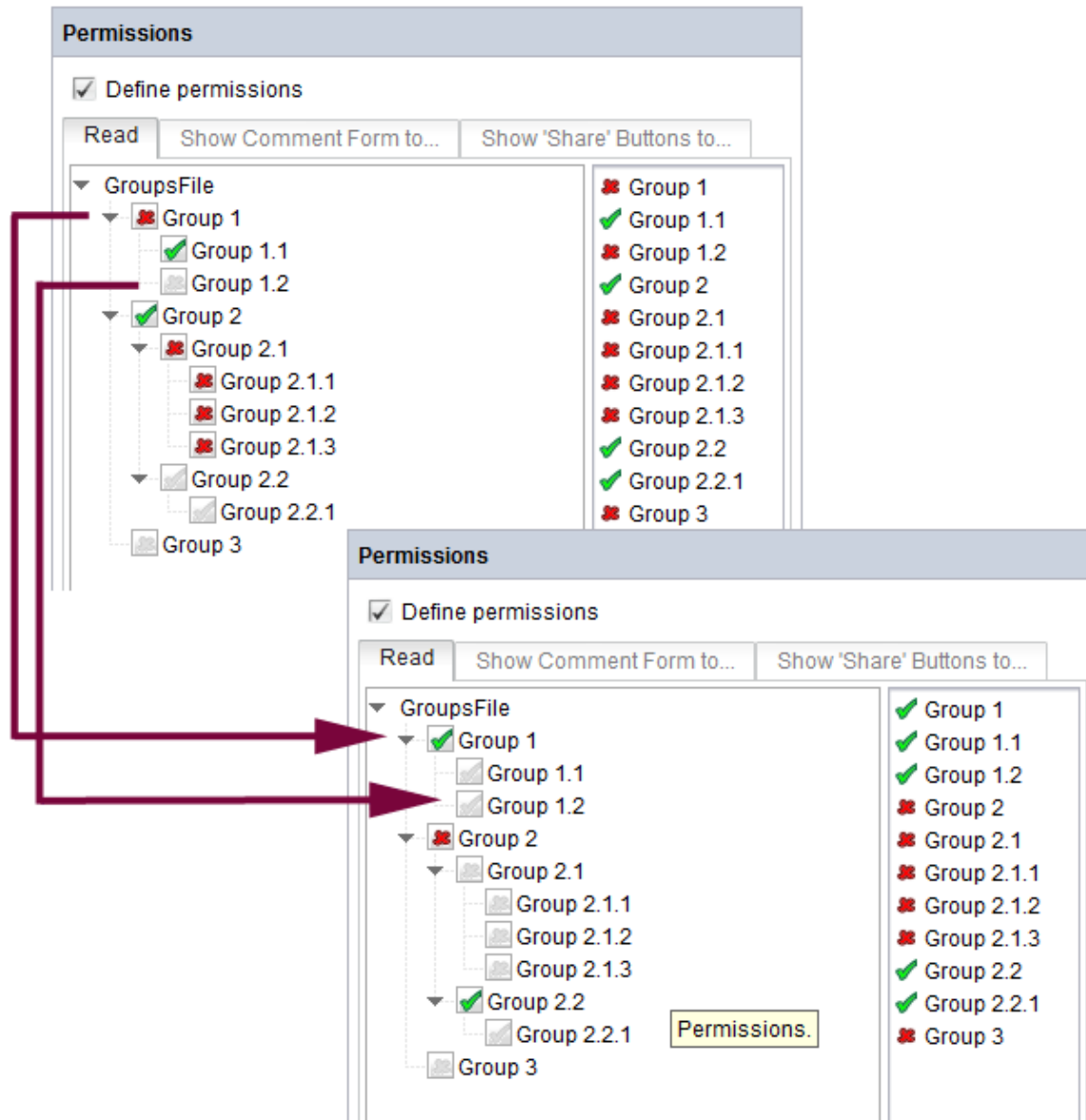
Contradictory permission assignments within a project can be uncovered using a script. The component supports the linking of scripts that can be executed in a different way, e.g.

- "on clicking", i.e. directly when a permission is defined in the component or
- via a button if a check is explicitly requested.

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 13-25: 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.

