

Administrationshandbuch

Imperia 8.6

Imperia AG
Von-der-Wettern-Straße 27
D - 51149 Köln
<http://www.imperia.net>

1. Einleitung	1
1.1. Konventionen in dieser Dokumentation	1
1.2. System-Architektur	1
1.3. Datenhaltung und Dokumentenmodell	2
1.4. Rollenkonzept	2
1.5. Rubriken	3
1.6. Workflow	3
1.7. IWE - Imperia-WYSIWYG-Editor	4
2. Installation von Imperia	5
2.1. Voraussetzungen	6
2.1.1. Dateisystem	6
2.1.2. Perl	6
2.1.3. Perl-Modul Storable	6
2.1.4. Perl-Modul DB_File	7
2.1.5. Perl-Modul BerkeleyDB	7
2.1.6. Perl-Modul DBI	7
2.1.7. Perl-Modul DBD::SQLite	7
2.1.8. ImageMagick	8
2.1.9. Webserver	8
2.1.10. Zusätzliche Voraussetzungen für SQL-basierte Datenhaltung	8
2.1.10.1. Perl-Module	8
2.1.10.2. Oracle®	8
2.1.10.3. MS-SQL®	8
2.1.10.4. MySQL	8
2.1.10.5. PostgreSQL	9
2.1.11. Weitere Informationen	9
2.2. Installation unter UNIX®-Derivaten	9
2.3. Installation unter Windows® 2000 / 2003 Server / XP	10
2.4. Installation eines Zielsystems	11
2.5. Installieren eines Patches ("Hotfix")	11
2.6. Update von einer älteren Imperia-Version	12
2.6.1. Referenzen	12
2.6.2. Updaten der Datenbankstruktur	12
2.7. Konfiguration von Imperia	12
2.7.1. ALLOW_TEMPLATE_FORM	12
2.7.2. ALLOW_TEMPLATE_SCRIPT	12
2.7.3. ALLOW_TEMPLATE_FRAME	12
2.7.4. UPLOAD_OBJECTS	13
2.7.5. OPERATING-SYSTEM	13
2.7.6. CGI-DIR	13
2.7.7. DOCUMENT-ROOT	13
2.7.8. CGI-BIN	13
2.7.9. DEMOMODE	13
2.7.10. SERVER-SCHEME	13
2.7.11. SERVER_NAME	13
2.7.12. SERVER-PORT	14
2.7.13. HTTP_HOST	14
2.7.14. LOCK_EX, LOCK_SH, LOCK_UN, LOCK_NB	14
2.7.15. COOKIE_TTL	14
2.7.16. SAFE_TEMPLATE	14
2.7.17. REG_KEY, REG_KEY2, REG_NAME	14
2.7.18. IMPERIA_ERROR_LOG	14
2.7.19. LATCHING	14
2.8. Modul-Installation	14
2.8.1. Verwendung des Grids	14
2.8.2. Hinweise für das JavaRuntime-Environment	15
2.9. Smart Database Update	15
2.9.1. Update und Wartung	15
2.9.1.1. site_doctree_admin.pl	15
2.9.1.2. site_doctree_repair.pl	16
3. Administration des Systems	17

3.1. Erstmaliges Starten und Anmelden	17
3.2. Wartungsmodi für Imperia	17
3.2.1. Wartungsmodus ohne Superuser-Zugriff	18
3.2.2. Wartungsmodus mit Superuser-Zugriff	18
3.3. Lizenzierung	19
3.4. Standard-Zielsystem	19
3.5. Systemdienste	20
3.5.1. Neuen Systemdienst anlegen	21
3.5.2. Systemdienst ändern	23
3.5.3. Systemdienst löschen	24
3.6. Modul-Lizenzierung	24
3.7. Zusätzliche Zielsysteme	25
3.7.1. Neuen Freischalt-Trigger eintragen	25
3.7.2. Freischalt-Trigger bearbeiten	26
3.7.3. Freischalt-Trigger löschen	27
3.7.4. Die Darstellung beim Freischalten ändern	27
3.8. Dokumentenhaltung	27
3.8.1. Unterschiede zum Dateisystem und Einschränkungen	27
3.8.2. Konfiguration in Imperia	28
3.8.2.1. File65	30
3.8.2.2. PostgreSQL	30
3.8.2.3. ODBC_MSSQL	30
3.8.2.4. Oracle	30
3.8.3. Konfiguration der Dokumentenhaltungsschnittstelle	31
3.8.3.1. Notwendige Perl-Module	31
3.8.3.2. PostgreSQL und Linux	32
3.8.3.3. PostgreSQL und Windows	33
3.8.3.4. Oracle und Linux	33
3.8.3.5. MySQL und Windows®	33
3.8.3.6. MySQL und Linux	34
3.8.3.7. SQL-Server und Windows®	34
3.8.4. Tabellen erzeugen	34
3.8.5. Tabellen löschen	35
3.9. One-Click-Edit	35
3.9.1. One-Click-Edit und Copy-Seiten	37
3.9.2. OCE-Skript hinzufügen	37
3.9.3. Imperia OCE-Skripte	37
3.9.3.1. oce_edit	37
3.9.3.2. oce_clone	38
3.9.3.3. oce_erase	38
3.9.3.4. oce_delete	38
3.9.3.5. oce_history	39
3.9.3.6. oce_info	39
3.9.3.7. oce_linkcheck	39
3.9.3.8. oce_reparse	39
3.9.3.9. oce_unpublish	40
3.10. Allgemeine Einstellungen	40
3.10.1. Konfiguration	40
3.10.2. Cookie-Lebensdauer	40
3.10.3. Dokumenten-History	40
3.10.4. Java-VM	40
3.10.5. Link-Checker	41
3.10.6. Superuser Authentifizierungs-Plug-In	41
3.10.7. System-Sprache	41
3.10.8. Template-Prozessor	41
3.10.9. Unicode in Imperia	41
3.11. Konfiguration des Systems über die Benutzeroberfläche	42
3.11.1. Konfiguration der Schnittstelle	44
3.12. Konfiguration der Link-Konsistenzprüfung	44
3.12.1. Konfigurationsbeispiel	47
3.13. Erweiterung des Schreibtisches	47

3.13.1. Schreibtischansichten einrichten	47
3.13.2. Schreibtisch-Erweiterungen in das System einbinden	48
3.13.3. Standard-Schreibtischansichten umgehen	49
3.14. Skinning der Flexmodul-Steuerelemente	50
3.14.1. FLEXMODULE-SKIN	50
3.14.2. FLEXINSTANCE-SKIN	51
3.15. Änderungsverfolgung	51
3.15.1. Änderung der Standardfarben	51
3.16. Reparsen umfangreicher Sites	52
3.16.1. Optionen	52
3.16.2. Argumente	53
3.16.3. Beispiel	53
3.17. Das interne Mail-System	54
4. Struktur	55
4.1. Rubriken	55
4.1.1. Neue Rubrik anlegen	55
4.1.2. Rubriken bearbeiten	56
4.1.2.1. Rubriken-Metainformationen	56
4.1.2.2. Rubriken-Metainformationen bearbeiten	57
4.1.2.3. Zugriff auf die Rubriken-Metainformationen	57
4.1.2.4. Verfügbare Sprachen für mehrsprachige Dokumente festlegen	58
4.1.3. Rubrikenrechte einstellen	58
4.1.4. Rubrikenreihenfolge ändern	59
4.1.5. Rubriken löschen	60
4.2. Metadateien	60
4.2.1. Neue Metadatei anlegen	61
4.2.2. Metadatei hochladen	61
4.2.3. Metadatei bearbeiten	62
4.2.4. Metadatei löschen	62
4.3. Templates	62
4.3.1. Neues Template anlegen	63
4.3.2. Template hochladen	64
4.3.3. Template-Vorschau	64
4.3.4. Template bearbeiten	64
4.3.5. Template löschen	64
4.4. Flexmodule	65
4.4.1. Neues Flexmodul erzeugen	66
4.4.2. Flexmodule hochladen	66
4.4.3. Flexmodule bearbeiten	67
4.4.4. Flexmodule löschen	67
4.5. Slotmodule	67
4.5.1. Neues Slotmodul erzeugen	68
4.5.2. Slotmodule hochladen	68
4.5.3. Slotmodule bearbeiten	69
4.5.4. Slotmodule löschen	69
4.6. Codeinclude-Dateien	69
4.6.1. Neues Codeinclude / Verzeichnis erstellen	70
4.6.2. Codeinclude-Dateien hochladen	70
4.6.3. Codeinclude-Dateien bearbeiten	72
4.6.4. Codeinclude-Dateien / Verzeichnisse löschen	72
4.7. Dynamische Module	73
4.7.1. Laden einer externen Datei in den Editor	73
4.8. Template-Reparser	73
4.9. Assets verschieben	75
4.10. Systemdateien abgleichen	75
4.11. Dateirechte in Imperia verwalten	75
5. IWE - Imperia-WYSIWYG-Editor	78
5.1. Migration vom EWE-Modul zum IWE	78
5.2. Konfiguration des IWE	78
5.3. Konfigurationsoptionen	79
5.3.1. Template-Only Optionen	79

5.3.2. Freie Optionen	80
5.3.2.1. Verhalten des Editors	80
5.3.2.2. Styles	81
5.3.2.3. HTML Ausgabe	83
5.3.2.4. Benutzer-Interface	83
6. Rubriken	85
6.1. Rubriken-Eigenschaften bestimmen	85
6.1.1. Rubriken-Informationen angeben	86
6.1.2. Workflow festlegen	86
6.1.3. Metadatei und Template bestimmen	87
6.1.4. Rubriken-Zugriff	87
6.1.4.1. Zugriffsrechte zuweisen	87
6.1.4.2. Zugriffsrecht aberkennen	88
6.1.5. Asset-Rubriken	88
6.1.5.1. Asset-Rubriken in 'Freigeschaltete Dokumente bearbeiten' anzeigen	88
7. User und Rollen	89
7.1. Neue Rolle anlegen	89
7.2. Rollendaten bearbeiten	91
7.3. Löschen einer Rolle	92
7.4. Neuen User anlegen	92
7.5. Userdaten bearbeiten	93
7.6. User löschen	94
7.7. LDAP und Verzeichnisdienste	94
7.7.1. Einschränkungen	94
7.7.2. Anbindung an einen Verzeichnisdienst	95
7.7.3. Verfahren zur Anmeldung an Imperia	95
7.7.4. Notwendige Konfiguration für LDAP	95
7.7.4.1. system.conf	95
7.7.4.2. ldap.conf	96
7.7.4.2.1. Informationen für den Verzeichnisdienst	96
7.7.4.2.2. Informationen über den LDAP-Zugang	96
7.7.4.2.3. Liste der User-Container	96
7.7.4.2.4. Liste der Attribute	96
7.7.4.2.5. Passwort-Verschlüsselung	97
7.7.4.2.6. Passwort-Kodierung	97
7.7.4.2.7. LDAP-Filter für gültige Imperia-User	98
7.7.4.2.8. Liste der Rollen-Container	98
7.7.4.2.9. Liste der Rollen-Attribute	98
7.7.4.2.10. LDAP-Filter zur Rollen-Identifikation	98
7.7.4.2.11. Gruppierungs-Schema	98
7.7.4.2.12. Superuser-Rolle	99
7.7.4.2.13. Superuser-Zugang	99
7.7.4.2.14. Benutzeranbindung	99
7.7.4.2.15. Schreibzugriff	99
7.7.4.3. Imperia als Frontend für einen LDAP-Verzeichnisdienst	100
7.7.4.3.1. RDN-Attribut eines neuen User-Eintrags	100
7.7.4.3.2. Objektklassen eines neuen Imperia-Users	100
7.7.4.3.3. Das RDN-Attribut eines neuen Rollen-Eintrags	100
7.7.4.3.4. Objektklassen einer neuen Imperia-Rolle	100
7.8. Authentifizierung über HTTP	100
7.8.1. Konfiguration	101
8. Workflows	102
8.1. Workflows verwalten	103
8.1.1. Workflow-Zugriffsrechte ändern	103
8.1.2. Neuen Workflow anlegen	103
8.1.3. Workflow bearbeiten	104
8.1.4. Workflow löschen	104
8.2. Das Grid	104
8.2.1. Plug-In-Leiste	105
8.2.2. Button-Leiste	105
8.2.3. Bedienung des Grids	105

8.2.3.1. Workflow-Plug-Ins	106
8.2.3.2. Workflow-Schritt anlegen	106
8.2.3.3. Eigenschaften eines Workflow-Schrittes einstellen	106
8.2.3.4. Workflow-Schritt im Arbeitsbereich verschieben	107
8.2.3.5. Workflow-Schritt duplizieren	107
8.2.3.6. Workflow-Schritt löschen	107
8.2.3.7. Workflow-Schritte verbinden	107
8.2.3.8. Verbindungen lösen	107
8.2.4. Zugriffsrechte für Workflow-Schritte	108
8.2.4.1. Zugriffsrechte einstellen	108
8.2.4.2. Zugriffsrechte entfernen	109
8.3. Workflow bearbeiten	109
8.4. Workflow löschen	109
8.5. Workflow-Plug-Ins	109
8.5.1. Asset-Aktionen	110
8.5.2. Annotate	111
8.5.3. Archiv-Bereiniger	111
8.5.4. Auto-Expiry	111
8.5.5. Auto-Publisher	111
8.5.6. Bearbeiten	112
8.5.7. Bearbeiten und Weiterleiten	113
8.5.8. Cat2DocMeta	113
8.5.9. Clone	114
8.5.10. DocSelector	116
8.5.11. Eraser	118
8.5.12. Exists	118
8.5.13. Expiry	118
8.5.14. Genehmigung	119
8.5.15. If-Else	119
8.5.16. Genehmigung mit Link-Check	119
8.5.17. Metaedit	120
8.5.18. Metasetter	121
8.5.19. MetaSplice	122
8.5.19.1. Mehrsprachige Workflows mit MetaSplice	122
8.5.20. Mülleimer	123
8.5.21. Multilanguage	123
8.5.22. Null	123
8.5.23. Publish	124
8.5.24. Shredder	124
8.5.25. SmartMeta	124
8.5.26. Unter-Workflow	124
8.5.27. Upload	125
8.5.28. Zufall	127
9. Menü-Management	128
9.1. Untermenü hinzufügen	128
9.1.1. Untermenü einem Ordner hinzufügen	129
9.2. Menüs bearbeiten	130

9.2.1. Menüpunkt zum Hauptmenü hinzufügen	130
9.2.2. Menüpunkt einem Untermenü hinzufügen	130
9.2.3. Zugriffsrechte für Menüs einstellen	130
9.2.4. Untermenü löschen	131
9.3. Menüstruktur anzeigen	131
10. Media-Asset-Management (MAM)	132
10.1. Mit Asset-Rubriken arbeiten	132
10.1.1. Asset-Rubrik anlegen	132
10.1.2. Asset-Rubrik bearbeiten	134
10.1.3. Asset-Rubrik löschen	134
10.1.4. Einzelne Assets und Asset-Rubriken verschieben	134
10.2. Import von MDB-Inhalten in das Media-Asset-Management	135
10.2.1. Durchführung des Imports	135
10.3. Import von Assets aus einem Ordner oder einer Zip-Datei	136
10.3.1. Vorarbeiten und Konfiguration	138
10.3.2. Import mit Hilfe des Kommandozeilentools	137
10.3.3. Import über die grafische Oberfläche	138
10.3.3.1. Vorarbeiten und Konfiguration	138
10.3.3.2. Die Web-Oberfläche	139
10.4. Konfiguration des Media-Asset-Managements	140
10.4.1. Globale Suchfilter definieren	140
10.4.2. Asset-Metafelder global verfügbar machen	140
10.4.3. Asset-Metafelder in der Listenansicht anzeigen lassen	140
10.4.4. Titel-Leiste des MAM-Rubrikenbaums festlegen	140
10.4.5. Sortierung des Asset-Rubrikenbaums festlegen	141
10.4.6. Template für die Erstellung von Asset-Rubriken festlegen	141
10.4.7. Standard-Anzahl für Mehrfachupload definieren	141
10.4.8. Mehrfachupload-Optionen einrichten	141
10.4.9. Linkanzeige der Listenansicht konfigurieren	141
10.4.10. Asset-Details fixieren	142
10.4.11. Felder verbergen	142
10.4.12. Datenhaltung für Binärdateien bestimmen	143
10.4.13. Datei-Benennung für Assets konfigurieren	143
10.4.14. Unabhängige Freischaltung von Assets aktivieren	144
10.4.15. Deaktivierung des Media-Asset-Managements	144
10.4.16. Verfügbare Asset-Aktionen nachträglich festlegen	145
10.4.17. Zugriffsrechte für Asset-Rubriken verwalten	145
10.4.18. Listen-Ansicht konfigurieren	145
10.4.18.1. Neue Spalten hinzufügen und löschen	145
10.4.18.2. Spaltenreihenfolge ändern	146
10.5. Referenzen	146
10.6. Einbau in ein Template	147
10.7. Mehrfachupload mit Flash	147
10.7.1. Kompatibilität	147
10.7.2. Einschränkungen	148
10.7.3. Rubrikeneinstellungen	148
11. Der Daemon Hermes	149
11.1. Starten des Hintergrund-Daemons Hermes	149
11.1.1. Starten des Daemons unter UNIX®	149
11.1.2. Starten des Daemons unter Windows®	149
11.1.3. Der Daemon als Systemdienst unter Windows®	149
11.1.3.1. Installation	150
11.1.3.2. Deinstallation	150
11.2. Kommandozeilenoptionen	150
11.2.1. Verwendung	150
11.2.2. Optionen	151
11.2.3. Log-Datei	152
11.2.4. Plug-In-Tabelle	152
11.3. Hermes-Import	153
11.3.1. Standard-Import-Konfiguration des Hermes-Daemons	153
11.3.2. Zielrubrik bestimmen	154

11.3.3. Vorhandene Dokumente beim Import mergen	155
11.3.4. Aufbau der Import-Daten	155
11.3.4.1. ASCII-Import	155
11.3.4.2. XML-Import	155
11.3.4.3. Meta-Import	156
12. Die Imperia-Volltextsuche	157
12.1. Einführung	157
12.2. Eingabe von Suchanfragen und Syntax für Suchbegriffe	157
12.2.1. Groß- und Kleinschreibung	157
12.2.2. Sonderzeichen und Interpunktions	157
12.2.3. Wildcards	157
12.2.4. Alternative Schreibweisen / Worttrennungen	158
12.2.5. Umlaute	158
12.2.6. Phonetisch ähnliche Alternativschreibweisen	158
12.2.7. Stopwörter	158
12.2.8. Phrasensuche	158
12.2.9. Kategorisierte Suchanfragen	158
12.2.10. URL-Suche	158
12.2.11. Wortstammsuche	159
12.2.12. Gruppierung	159
12.2.13. Nach Metafeldinhalten filtern	159
12.2.14. Suchwörter verknüpfen und ausschließen, logische Operatoren	159
12.3. Anzeige von Suchergebnissen	159
12.3.1. Kontextanzeige	160
12.3.2. Highlighting	160
12.3.3. Ergebnisfilter	160
12.4. Berechnung der Treffer-Relevanz und Boni	160
12.5. Einrichten der Imperia-Volltextsuche	160
12.5.1. Voraussetzungen	161
12.5.2. Schnellstartanleitung	161
12.5.3. Konfiguration der Volltextsuche	162
12.5.3.1. Allgemeine Syntaxkonventionen	162
12.5.3.2. Reguläre Ausdrücke in der Konfiguration	163
12.5.3.3. Die Konfiguration der Indexerstellung (index.conf)	163
12.5.3.3.1. Basis-Tags und -Direktiven	164
12.5.3.3.2. Tags in Domain-Blöcken	168
12.5.3.3.3. Direktiven in Exclude-Blöcken	171
12.5.3.3.4. Direktiven in Include-Blöcken	171
12.5.3.3.5. Tags und Direktiven in Directory-Blöcken	172
12.5.3.3.6. Option-Direktiven	176
12.5.3.3.7. Tags und Direktiven in MetaGroup-Blöcken	179
12.5.3.3.8. Dynamisch benannte Meta-Gruppen	180
12.5.3.4. Die Konfiguration der Suche (fts.conf)	181
12.5.3.4.1. Basis-Tags	181
12.5.3.4.2. Tags in Domain-Blöcken	182
12.5.3.4.3. Direktiven in Option-Blöcken	185
12.5.3.4.4. Direktiven in parseroption-Blöcken	189
12.5.3.4.5. Direktiven in Output-Blöcken	190
12.5.3.4.6. UserSort: Sortierung der Treffer bestimmen	191
12.5.3.4.7. DateBonus: Treffer mit datumsbasierten Boni sortieren	192
12.5.3.5. Indexerstellung	193
12.5.3.6. Erstellung von Suchskripten	194
12.5.3.7. Stopwortlisten anlegen und verwalten	194
13. Anhang	197
13.1. Beispiele für LDAP.conf	197
13.1.1. PAM	197
13.1.2. ADIR	198
13.2. Dokumentenhaltung in einer Datenbank	198
13.2.1. Aufbau der Imperia-Tabellen	198
13.2.1.1. Attribute in den Imperia-Tabellen	199
13.2.1.1.1. Tabelle archive_meta	200

13.2.1.1.2. Tabelle category	200
13.2.1.1.3. Tabelle category_acl	200
13.2.1.1.4. Tabelle category_meta	201
13.2.1.1.5. Tabelle currentfile	201
13.2.1.1.6. Tabelle desktop	202
13.2.1.1.7. Tabelle document	202
13.2.1.1.8. Tabelle document_acl	202
13.2.1.1.9. Tabelle history	203
13.2.1.1.10. Tabelle image	203
13.2.1.1.11. Tabelle longlive	203
13.2.1.1.12. Tabelle meta	204
13.2.1.1.13. Tabelle object	204
13.2.1.1.14. Tabelle sequence_table	204
13.2.1.1.15. Tabelle ticket_data	204
13.2.1.1.16. Tabelle version	204
13.3. Die Imperia Admin-Skripte	205
13.3.1. admin_accesscat.pl	205
13.3.2. admin_accessmeta.pl	205
13.3.3. admin_accesstemplate.pl	205
13.3.4. admin_accessworkflow.pl	205
13.3.5. admin_acl_update.pl	205
13.3.6. admin_addcat.pl	205
13.3.7. admin_adddoc.pl	205
13.3.8. admin_addrole.pl	206
13.3.9. admin_adduser.pl	206
13.3.10. admin_cleandesktop.pl	206
13.3.11. admin_currentfile.pl	206
13.3.12. admin_delrole.pl	206
13.3.13. admin_deluser.pl	206
13.3.14. admin_deriverole.pl	206
13.3.15. admin_expire.pl	206
13.3.16. admin_getcat.pl	206
13.3.17. admin_getdoc.pl	207
13.3.18. admin_getrole.pl	207
13.3.19. admin_getuser.pl	207
13.3.20. admin_grpaccess.pl	207
13.3.21. admin_grpuser.pl	207
13.3.22. admin_listcat.pl	207
13.3.23. admin_listflex.pl	207
13.3.24. admin_listmeta.pl	207
13.3.25. admin_listrole.pl	207
13.3.26. admin_listtemplate.pl	207
13.3.27. admin_listuser.pl	207
13.3.28. admin_listworkflow.pl	207
13.3.29. admin_sanitisearchive.pl	207
13.3.30. admin_searchdoc.pl	208
13.3.31. admin_searchuser.pl	208
13.3.32. admin_setcat.pl	208
13.3.33. admin_setcounter.pl	208
13.3.34. admin_setdoc.pl	208
13.3.35. admin_setrole.pl	208
13.3.36. admin_setuser.pl	208
13.3.37. admin_statdoc.pl	208
13.4. Liste aller Variablen zur System-Konfiguration	209
13.4.1. Gruppe ACLs	209
13.4.1.1. Variable ENABLE_FLEX_ACL	209
13.4.2. Gruppe Anzeige	209
13.4.2.1. Variable BOTTOMTABLE	209
13.4.2.2. Variable CATEGORY_DELETION_PROMPT	209
13.4.2.3. Variable DISABLE-DATE-TOOLTIP	209
13.4.2.4. Variable DISABLE-FAV-MENU	209

13.4.2.5. Variable DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER.....	209
13.4.2.6. Variable DISABLE_STREAMED_OUTPUT	209
13.4.2.7. Variable DISABLE_TOOLTIP	210
13.4.2.8. Variable DISPLAY_BASE_CLASS	210
13.4.2.9. Variable DONT_ASK_BEFORE_EXPLODE_ALL_NODES	210
13.4.2.10. Variable FAV_MENU_LINK_ONLY	210
13.4.2.11. Variable FROMLIVE_DELETE_DEFAULT_LIVE	210
13.4.2.12. Variable LINKTOOL_DISPLAY_ASSETS	210
13.4.2.13. Variable GRID_HEIGHT	210
13.4.2.14. Variable GRID_WIDTH	210
13.4.2.15. Variable IBS_PAGE	210
13.4.2.16. Variable IMPERIA_SKIN_PATH	210
13.4.2.17. Variable LATCHING	210
13.4.2.18. Variable MAIN_MENU_LINK_ONLY	211
13.4.2.19. Variable MD-WINDOW-HEIGHT	211
13.4.2.20. Variable MD-WINDOW-WIDTH	211
13.4.2.21. Variable NO-NPH	211
13.4.2.22. Variable POPUP_WINDOW_HEIGHT	211
13.4.2.23. Variable POPUP_WINDOW_WIDTH	211
13.4.2.24. Variable TOPTABLE	211
13.4.3. Gruppe Authentifizierung	211
13.4.3.1. Variable AUTH_CHAINED_PLUGINS	211
13.4.3.2. Variable AUTH_PLUGIN	211
13.4.3.3. Variable AUTH_PLUGIN_BACKUP	211
13.4.3.4. Variable COMMWORLD_AUTH_CHECKSESSION	211
13.4.4. Gruppe Datenhaltungs-Plug-In Datei	212
13.4.4.1. Variable FILE65-DOCUMENT-DIR	212
13.4.5. Gruppe Dynamische Variablen	212
13.4.5.1. Variable ABS-CGI-BIN	212
13.4.5.2. Variable ABS-DOC-ROOT	212
13.4.6. Gruppe Exporter	212
13.4.6.1. Variable XALAN_CLASSPATH	212
13.4.6.2. Variable XSLTPROC_PATH	212
13.4.6.3. Variable XSLT_PLUGIN	212
13.4.7. Gruppe File Transfer	212
13.4.7.1. Variable DISABLE_WRITE_TRANSMISSION_CACHE	212
13.4.7.2. Variable OPEN_3_USE_SYSREAD	212
13.4.7.3. Variable SSH-OPTIONS	213
13.4.7.4. Variable SSH-CLIENT	213
13.4.7.5. Variable SSH-FILED	213
13.4.8. Gruppe Flexmodule	213
13.4.8.1. Variable ALLOW_PERL_UPLOAD	213
13.4.8.2. Variable FLEXMODULE-SKIN	213
13.4.8.3. Variable FLEXINSTANCE-SKIN	213
13.4.9. Gruppe Grundeinstellungen	213
13.4.9.1. Variable CGI-BIN	213
13.4.9.2. Variable CGI-DIR	213
13.4.9.3. Variable DEMOMODE	213
13.4.9.4. Variable DISABLE-FLOCK	213
13.4.9.5. Variable DISABLE-FORTUNE-COOKIE	213
13.4.9.6. Variable DOCTREE_PLUGIN	214
13.4.9.7. Variable COUNTER-DIR	214
13.4.9.8. Variable DEBUG_ArchiveCleaner	214
13.4.9.9. Variable ARCHIVESHREDDER	214
13.4.9.10. Variable ARCHIVE_DISABLE_DELETE	214
13.4.9.11. Variable DISGUISE_JAVA_PLUGIN	214
13.4.9.12. Variable FLOCK_DEBUG	214
13.4.9.13. Variable FollowSymLinks	214
13.4.9.14. Variable IMPERIA-VERSION	214
13.4.9.15. Variable IMPERIA_ERROR_LOG	214
13.4.9.16. Variable MAX_REQUEST_SIZE	214

13.4.9.17. Variable MIME_TYPES	215
13.4.9.18. Variable USE_STRICT	215
13.4.9.19. Variable PACK_STR	215
13.4.9.20. Variable SECRET_FREQUENCY	215
13.4.9.21. Variable SECURE_UMASK	215
13.4.9.22. Variable UMASK	215
13.4.9.23. Variable DOCUMENT_ROOT	215
13.4.9.24. Variable OPERATING_SYSTEM	215
13.4.9.25. Variable SITE-DIR	215
13.4.10. Gruppe HTTP Cookie	215
13.4.10.1. Variable COOKIE-DOMAIN	215
13.4.10.2. Variable COOKIE-PATH	216
13.4.10.3. Variable COOKIE_NAME	216
13.4.10.4. Variable COOKIE_TTL	216
13.4.11. Gruppe Hermes	216
13.4.11.1. Variable HERMES_RECONNECT	216
13.4.11.2. Variable MAX_AUTO_PUBLISH_DOCS	216
13.4.11.3. Variable HERMES_INCOMING_DELAY	216
13.4.12. Gruppe History	216
13.4.12.1. Variable DEFAULT_HISTORY_LENGTH	216
13.4.12.2. Variable HISTORY_HIDE_DATE	216
13.4.12.3. Variable HISTORY_HIDE_DATE_SU	216
13.4.13. Gruppe I18N	217
13.4.13.1. Variable DEFAULT_LOCALE	217
13.4.13.2. Variable LANGUAGE	217
13.4.13.3. Variable SYSTEM-LANGUAGE	217
13.4.13.4. Variable SYSTEM_LOCALE	217
13.4.13.5. Variable USER_LOCALE	217
13.4.14. Gruppe Imperia Develop Session	217
13.4.14.1. Variable ADDR_AUTH	217
13.4.14.2. Variable CRYPT_ALGO	217
13.4.15. Gruppe Datum und Zeit	217
13.4.15.1. Variable CALENDAR_DATE_FORMAT	217
13.4.15.2. Variable CALENDAR_TIME_FORMAT	217
13.4.15.3. Variable DATE_COMBO_MINUTES_STEP	217
13.4.15.4. Variable DATE_COMBO_YEARS_LIMIT	218
13.4.15.5. Variable DISPLAY_TIME_ZONE	218
13.4.16. Gruppe IWE	218
13.4.16.1. Variable EWE_COMPAT_MODE	218
13.4.16.2. Variable TEMPLATE_EWE	218
13.4.17. Gruppe Link-Checker	218
13.4.17.1. Variable LINK_CHECK	218
13.4.17.2. Variable NOPROXY	218
13.4.17.3. Variable PROXY_FTP	218
13.4.17.4. Variable PROXY_HTTP	218
13.4.18. Gruppe Lock Constants	219
13.4.18.1. Variable LOCK_EX	219
13.4.18.2. Variable LOCK_NB	219
13.4.18.3. Variable LOCK_SH	219
13.4.18.4. Variable LOCK_UN	219
13.4.19. Gruppe MDB Counter	219
13.4.19.1. Variable DISABLE_MDB_REFCOUNT	219
13.4.19.2. Variable LIVE_CLEANUP	219
13.4.19.3. Variable REINDEX_MDB_FREQUENCY	219
13.4.20. Gruppe Media-Asset-Management	219
13.4.20.1. DISABLE_8BIT_FILENAME	219
13.4.20.2. Variable MAM_CAT_ORDER_BY_NAME	220
13.4.20.3. Variable MAM_CATEGORY_TEMPLATE	220
13.4.20.4. Variable MAM_DISABLE	220
13.4.20.5. Variable MAM_ENABLE_PUBLISH	220
13.4.20.6. Variable MAM_GET_UPLOAD_FILENAME	220

13.4.20.7. Variable MAM_HIDDEN_FIELDS	220
13.4.20.8. MAM_JPEG_QUALITY	221
13.4.20.9. Variable MAM_LINKS_STYLE	221
13.4.20.10. Variable MAM_MULTIUPLOAD_OPTIONS	222
13.4.20.11. Variable MAM_MULTIUPLOAD_SIZE	222
13.4.20.12. Variable MAM_NAV_HEADER	222
13.4.20.13. Variable MAM_NO_EXTENDER_NORMALIZE	222
13.4.20.14. Variable MAM_PUBLIC_SPOOL	222
13.4.20.15. Variable MAM_READ_ONLY_FIELDS	222
13.4.20.16. Variable MAM_SAVED_FILTERS	222
13.4.20.17. MAM_SPOOL_CHARSET	222
13.4.20.18. Variable MAM_SPOOL_PATH	223
13.4.20.19. Variable MAM_UNIQUE	223
13.4.21. Gruppe Mediendatenbank	223
13.4.21.1. Variable MD-BROWSER-VIEW	223
13.4.21.2. Variable MD-PICS-PER-ROW	223
13.4.21.3. Variable MD_ENABLE_RECURSIVE	223
13.4.21.4. Variable MIME_ICON_DIR	223
13.4.21.5. Variable MIME_TXT_DEFAULT_EXTENDER	223
13.4.21.6. Variable UPLOAD_MDB	223
13.4.22. Gruppe One-Click-Edit	224
13.4.22.1. Variable ONECLICKEDIT	224
13.4.23. Gruppe PHP	224
13.4.23.1. Variable PHP_CLI	224
13.4.23.2. Variable PHP_PATH	224
13.4.24. Gruppe Publisher	224
13.4.24.1. Variable DEFAULT_VERTOSITY	224
13.4.24.2. Variable DISABLE_PUBLISHER_WARNINGS	224
13.4.24.3. Variable INFORM_ROLE_ON_PUBLISH	224
13.4.24.4. Variable SERIALIZER	224
13.4.24.5. Variable UPLOAD_OBJECTS	224
13.4.25. Gruppe Registrierungsschlüssel	224
13.4.25.1. Variable REG_DEMO	224
13.4.25.2. Variable REG_KEY	225
13.4.25.3. Variable REG_NAME	225
13.4.26. Gruppe Reparsen	225
13.4.26.1. Variable REPARSE_ACTION_CHECKED	225
13.4.26.2. Variable REPARSE_ACTION_FIXED	225
13.4.26.3. Variable REPARSE_SOURCE_CHECKED	225
13.4.26.4. Variable REPARSE_SOURCE_FIXED	225
13.4.27. Gruppe Rubriken	225
13.4.27.1. Variable CATEGORY_TEMPLATE	225
13.4.27.2. Variable DEFAULT_DIRECTORY	225
13.4.27.3. Variable DEFAULT_FILENAME	225
13.4.28. Gruppe Schreibtisch	225
13.4.28.1. Variable DESKTOP_COLUMN_MODULES	225
13.4.28.2. Variable DESKTOP_CONFIG_MODULES	225
13.4.29. Gruppe Server	226
13.4.29.1. Variable SERVER-SCHEME	226
13.4.29.2. Variable SERVER-URL	226
13.4.29.3. Variable SERVER_NAME	226
13.4.30. Gruppe SiteActive	226
13.4.30.1. Variable SAVEPAGE-PASSWORD	226
13.4.30.2. Variable SCHEDULE	226
13.4.31. Gruppe Software Key	226
13.4.31.1. Variable UCROP_KEY	226
13.4.31.2. Variable WORDMODUL_CLASSID	226
13.4.31.3. Variable WORDMODUL_KEY	226
13.4.31.4. Variable WORDMODUL_USER	226
13.4.32. Gruppe Standard-Zielsystem	226
13.4.32.1. Variable live-url	226

13.4.32.2. Variable liveftp1-directory	226
13.4.32.3. Variable liveftp1-ip	226
13.4.32.4. Variable liveftp1-name	226
13.4.32.5. Variable liveftp1-pass	227
13.4.32.6. Variable liveftp1-site	227
13.4.32.7. Variable liveftp1_passive_mode	227
13.4.32.8. Variable transfer	227
13.4.33. Gruppe Template	227
13.4.33.1. Variable ALLOW_TEMPLATE_FORM	227
13.4.33.2. Variable ALLOW_TEMPLATE_FRAME	227
13.4.33.3. Variable ALLOW_TEMPLATE_SCRIPT	227
13.4.33.4. Variable MD-DEFAULT-LINKS	227
13.4.33.5. Variable NO_ANON	227
13.4.33.6. Variable SAFE_TEMPLATE	227
13.4.33.7. Variable TEMPLATE_EWE	227
13.4.34. Gruppe Templateprozessor	227
13.4.34.1. Variable TEMPLATE_CHAIN	227
13.4.35. Gruppe User-Management-Einstellungen	228
13.4.35.1. Variable DATABASE_STORAGE_PLUGIN	228
13.4.35.2. Variable SORT_ROLES	228
13.4.35.3. Variable TRANSITIVE-ACCESS	228
13.4.35.4. Variable UMLIGHT_PLUGIN	228
13.4.36. Gruppe Workflow	228
13.4.36.1. Variable CONFIRM_NEXT_STEP	228
13.4.36.2. Variable APPROVAL_DISABLE_JUMP	228
Index	229

Abbildungsverzeichnis

1.1. Imperia-Architektur	1
3.1. Login-Maske	17
3.2. Wartungsmodus	18
3.3. Dialog Registrierungsschlüssel	19
3.4. Dialog Standard-Zielsystem	19
3.5. Dialog zum Anlegen eines Systemdienstes	22
3.6. Dialog Neuen Systemdienst anlegen	22
3.7. Dialog Zusätzliche Zielsysteme	25
3.8. Neues Zielsystem hinzufügen	25
3.9. Dialog Dokumentenhaltung	28
3.10. Dialog OCE-Konfiguration	36
3.11. Übersicht der editierbaren Konfigurationsdateien	42
3.12. Anzeige des Inhalts einer Konfigurationsdatei	43
3.13. Der Editor für Konfigurationsdateien	43
3.14. Die Datei config.conf	44
4.1. Dialog Rubriken bearbeiten	55
4.2. Rubriken-ACL-Verwaltung	59
4.3. Dialog Metadateien bearbeiten	60
4.4. Dialog Neue Metadatei anlegen	61
4.5. Dialog Metadatei hochladen	62
4.6. Dialog Templates bearbeiten	63
4.7. Dialog Neues Template anlegen	63
4.8. Dialog Template hochladen	64
4.9. Dialog Flexmodule bearbeiten	65
4.10. Dialog Neues Flexmodul erzeugen	66
4.11. Dialog Flexmodul hochladen	66
4.12. Dialog Slotmodule bearbeiten	67
4.13. Dialog Neues Slotmodul erzeugen	68
4.14. Dialog Slotmodul hochladen	68
4.15. Der Dialog Codeincludes bearbeiten	69
4.16. Der Dialog Neue Codeinclude-Datei anlegen	70
4.17. Der Dialog Codeinclude-Datei hochladen	71
4.18. Das Bearbeitungsfenster für Codeincludes	72
4.19. Dialog Template-Reparser	74
4.20. Dialog Rechtevergabe, Beispiel Metadateien	76
7.1. Dialog Globale Rechte für ACL-gesteuerte Objekte	90
7.2. Dialog Rollenauswahl	91
7.3. Dialog Userauswahl	93
8.1. Dialog Ändern der Workflow-Zugriffsrechte	103
8.2. Dialog Workflow anlegen	104
8.3. Das Grid	105
8.4. Dialog Workflow-Schritt Eigenschaften (Edit-Plug-In)	107
8.5. Konfiguration des Plug-Ins Asset-Aktionen	110
8.6. Konfiguration des Bearbeiten-Plug-Ins	112
8.7. Beispiel Bearbeiten & Weiter-Plug-In	113
8.8. Konfiguration des Plug-Ins Cat2DocMeta	114
8.9. Konfiguration des Plug-Ins Clone	115
8.10. Eigenschaften des IfElse-Plug-Ins	119
8.11. Dialog Link-Checker	120
8.12. Eigenschaften des Metaedit-Plug-Ins	121
8.13. Eigenschaften MetaSetter	122
8.14. Eigenschaften des Plug-Ins MetaSplice	122
8.15. mehrsprachiger Workflow mit MetaSplice	123
8.16. Konfiguration von MetaSplice	123
8.17. Konfiguration von If-Else	123
8.18. Beispiel-Workflow ohne Unter-Workflows	124
8.19. Der Unter-Workflow	125
8.20. Der abgeänderte Haupt-Workflow	125
8.21. Konfiguration des Unter-Workflows	125

8.22. Konfiguration des Upload-Plug-Ins	126
8.23. Warnmeldung bei Dateinamensgleichheit	127
10.1. Der Dialog Assets verschieben mit ausgeklappter Rubrikenstruktur	134
10.2. Upload-Feld mit Möglichkeit zur Eingabe des Dateinamens	143
10.3. Der Dialog Neue Spalte	145
10.4. Der Dialog Anpassen	146
10.5. Dialog Asset-Referenzen	147
11.1. Beispiel Konfiguration der XYNTService.ini	150

Tabellenverzeichnis

1.1. Schlüssel-Wert-Paare eines neuen Dokuments	2
3.1. SQL-Datenbanken für die Dokumentenhaltung	27
3.2. RDBMS spezifische Treiber	31
3.3. CPAN-Module	32
10.1. Parameter für <code>mam_import_mdb.pl</code>	135
10.2. Verfügbare Felder im Media-Asset-Management	142
12.1. Shortcuts für logische Operatoren	159
12.2. Einstellungen der Direktive <code>verbosityLevel</code>	179
12.3. allgemein verfügbare Metafelder	179
12.4. Einstellungen für <code>queryMode</code> : verschiedene Suchmodi	187
12.5. Sortiermethoden	191
12.6. Zeiteinheiten für die Direktive <code>bonus_date_period</code>	192
12.7. Angaben für die Variable <code>bonus_date_field</code>	193
12.8. Einstellungen der Direktive <code>verbosityLevel</code>	193
12.9. Parameter für <code>site/bin/fts_blacklist.pl</code>	195
13.1. Datenbank-Tabellen für Imperia	199
13.2. Tabelle <code>archive</code>	200
13.3. Tabelle <code>category</code>	200
13.4. Tabelle <code>category_acl</code>	200
13.5. Tabelle <code>category_meta</code>	201
13.6. Tabelle <code>currentfile</code>	201
13.7. Tabelle <code>desktop</code>	202
13.8. Tabelle <code>document</code>	202
13.9. Tabelle <code>document_acl</code>	202
13.10. Tabelle <code>history</code>	203
13.11. Tabelle <code>longlive</code>	203
13.12. Tabelle <code>meta</code>	204
13.13. Tabelle <code>object</code>	204
13.14. Tabelle <code>sequence_table</code>	204
13.15. Tabelle <code>ticket_data</code>	204
13.16. Tabelle <code>version</code>	204
13.17. Verfügbare Felder im Media-Asset-Management	221

Kapitel 1. Einleitung

Dieses Kapitel gibt einen Überblick über die Funktionsweise und die Systemarchitektur der Imperia-Software. Das System funktioniert aufgrund verschiedener Konzepte und Komponenten, die wir in diesem Kapitel behandeln. Hierzu gehören:

- System-Architektur
- Rollenkonzept
- Workflow
- Datenhaltung und Dokumentenmodell

1.1 Konventionen in dieser Dokumentation

Die vorliegende Dokumentation benutzt folgende Konventionen:

- Dialoge, Abschnitte, Eingabefelder, Markierungsfelder, kurz alle Programmbestandteile werden fett gedruckt dargestellt.
- Auf besondere Zusammenhänge wird mit der fett gedruckten Überschrift **Achtung:** hingewiesen.
- Hinweise werden kursiv dargestellt.
- Platzhalter innerhalb von Code-Beispielen werden kursiv dargestellt.

1.2 System-Architektur

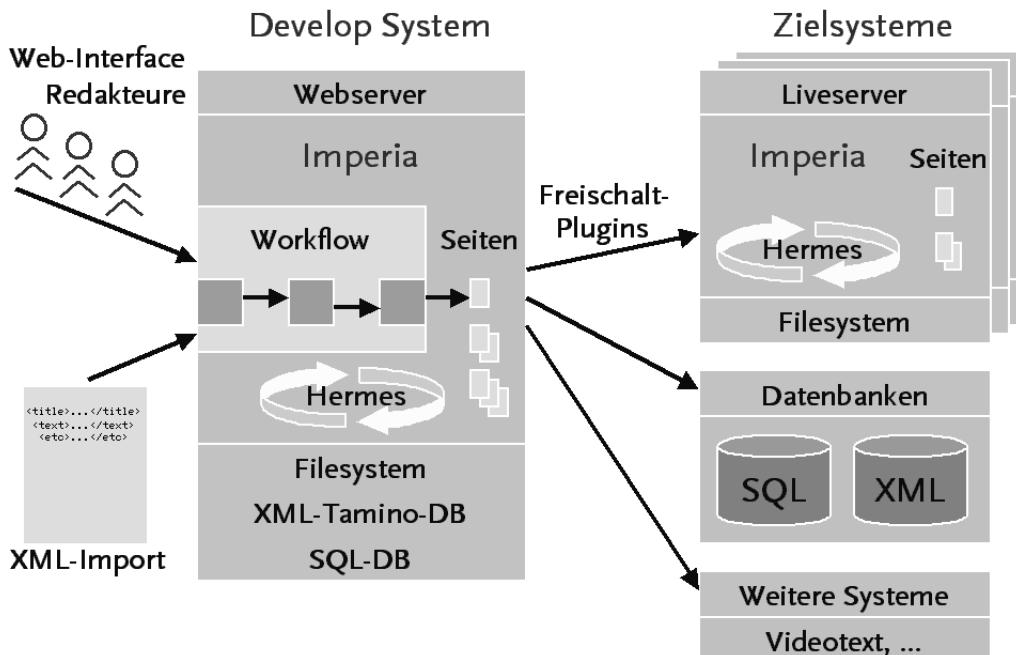


Abb. 1.1: Imperia-Architektur

Imperia besteht im Wesentlichen aus zwei Systemen: dem Produktionssystem (historische Synonyme: Develop-System oder Develop) und den Zielsystemen (historische Synonyme: Live-Systeme oder Live).

Das Produktionssystem ist der Teil von Imperia, in dem Dokumente von Redakteuren erzeugt und bearbeitet werden. Welche Arbeiten im Einzelnen an den Dokumenten durchgeführt werden, hängt unter anderem sehr eng mit den im Produktionssystem angelegten Rollen und Workflows zusammen.

Ist die Arbeit an einem Dokument erledigt, wird es, entweder manuell oder automatisch, vom Produktionsystem auf die Zielsysteme übertragen. Diesen Vorgang nennt man **freischalten**. Das Freischalten geschieht über Freischalt-Plug-Ins, mit deren Hilfe verschiedene Zielformate auf dem Zielsystem erzeugt werden können. Verwendet man das HTML-Freischalt-Plug-In, so kopiert Imperia die schon auf dem Produktionsystem im Filesystem abgelegten und über Templates generierten Dokumente in das Filesystem des entsprechenden Zielsystems. Aus diesen freigeschalteten Dokumenten kann das Zielsystem automatisch Übersichtsseiten generieren und so dafür sorgen, dass diese immer einen aktuellen Überblick über die Themen des Webauftritts bieten. Diese Aufgabe übernimmt **SiteActive** in Zusammenarbeit mit dem Hintergrund-Daemon **Hermes** und dem Notification-Freischalt-Plug-In. Durch dieses Notification-Plug-In werden alle Metainformationen eines Dokumentes in einem Binärformat auf das Zielsystem kopiert. Der Hintergrund-Daemon **Hermes** verarbeitet dieses **Notification-File** und legt auf dem Zielsystem der Imperia-Installation eine **Datenhaltung** an. Zusätzlich werden über das Notification-File die sogenannten Systemdienste getriggert, die die oben erwähnten **SiteActive**-Funktionen steuern.

Mit dem XML- und SQL-Freischalt-Plug-In stehen weitere Zielformate zur Verfügung, die den in Imperia verwalteten Inhalt strukturiert ausgeben. Darüber hinaus können eigene Freischalt-Plug-Ins entwickelt werden.

1.3 Datenhaltung und Dokumentenmodell

Imperia bietet verschiedene Möglichkeiten der Datenhaltung. Einmal kann man die Daten im Dateisystem speichern. Diese Möglichkeit ist bei kleiner und mittlerer Projektgröße sehr performant. Eine weitere Möglichkeit der Dokumentenhaltung ist eine SQL-Datenbank. Imperia unterstützt eine Reihe von kommerziellen und freien SQL-Datenbanken.

Nach der Installation speichert Imperia die Dokumente im Dateisystem. Wenn eine andere Möglichkeit der Datenhaltung verwendet werden soll, sollte dies eingestellt werden, bevor Dokumente erzeugt werden. Aber auch ein nachträgliches Umschalten der Datenhaltung ist möglich - dazu stellt Imperia entsprechende Admin-Skripte zur Verfügung.

Ein Imperia-Dokument ist eine Sammlung von Informationen. Neben dem eigentlichen Dokumenteninhalt gehören dazu auch Angaben, die das Dokument beschreiben, beispielsweise der Verfasser und das verwendete Template. Imperia speichert sämtliche Informationen eines Dokuments in so genannten Metafeldern. Diese sind als Schlüssel-Wert-Paare aufgebaut, wobei der Schlüssel den Namen des betreffenden Feldes festlegt und der Wert dessen Inhalt. Jeder Schlüssel steht für eine Meta-Variable, jeder Wert für den Inhalt der Meta-Variable.

Beim Erzeugen eines Dokuments werden bereits einige Metafelder angelegt und gefüllt, so dass ein gerade erzeugtes Dokument unter anderem die folgenden Schlüssel-Wert-Paare besitzt:

Schlüssel	Beispielwert
directory	/news/aktuell
template	myTemplate
metafile	mymeta

Tabelle 1.1. Schlüssel-Wert-Paare eines neuen Dokuments

Diese Schlüssel und Werte erbt das Dokument von den Rubriken-Eigenschaften der Rubrik, die der User beim Erzeugen des neuen Dokuments ausgewählt hat. Jedes der Schlüssel-Wert-Paare kann jedoch an jeder Stelle des Workflows verändert werden, zum Beispiel durch ein Workflow-Plug-In (Metasetter) oder durch einen User in einem Edit-Schritt.

1.4 Rollenkonzept

Ein wesentlicher Bestandteil des Produktionssystems sind Rollen. Rollen verleihen einem User Rechte auf Imperia-Module, zum Beispiel auf das Media Asset Management (MAM) oder das Archiv, und auf Programmfunctionen, wie zum Beispiel **Neues Dokument erzeugen** oder **Dokumente freischalten**.

Ferner verleihen Rollen dem User Rechte auf Rubriken, die bestimmte Bereiche des Projekts darstellen. Er kann dann nur in diesen Projektbereichen die Funktionen ausüben, die ihm seine Rollen verleihen.

Rollen werden häufig so angelegt, dass jede Rolle bestimmte Funktionen oder Zugriff auf bestimmte Rubriken verleiht. Jeder User kann gleichzeitig mehreren Rollen angehören, so dass er die Summe der Rechte aller angenommenen Rollen erhält. Die Kombination mehrerer Rollen gibt einem User also die Rechte, die er benötigt, um die für ihn vorgesehenen Funktionen in den für ihn vorgesehenen Projektbereichen auszuführen.

In einer Hierarchie von Rollen erben Unterrollen die Rechte der übergeordneten Rollen. Aufgrund dessen ist es meist sinnvoll, übergeordnete Rollen nur mit allgemeinen Rechten auszustatten und die spezifischeren Rechte den Unterrollen zu zuweisen.

Wäre beispielsweise eine Rolle **Redakteur** vorhanden, die Rechte auf redaktionelle Funktionen (**Neues Dokument erzeugen, Archiv, Schreibtisch**, etc.) verleiht, könnten Unterrollen angelegt werden, die Zugriff auf Rubriken verleihen. Ein User darf dann sowohl die Rolle **Redakteur** als auch eine oder mehrere Unterrollen für seine Rubriken annehmen.

Weitere Informationen finden Sie in Kapitel 7 **User und Rollen**.

1.5 Rubriken

Die unterschiedlichen Themengebiete eines Projektes lassen sich in Imperia durch Rubriken repräsentieren. Diese lassen sich jeweils in mehrere Unterrubriken gliedern. Dabei sind die Unterrubriken bei der Erstellung mit den Einstellungen der übergeordneten Rubriken vorbelegt.

Über die Imperia-Rechteverwaltung legen Sie fest, wer in den einzelnen Rubriken lesen und schreiben bzw. Dokumente erstellen darf. Damit ein Benutzer in einer Unterrubrik Dokumente erzeugen kann, muss er in allen Rubriken, die im Ast über dieser Rubrik liegen, mindestens Leserechte haben.

Jedes Dokument, das in einer Rubrik erzeugt wird, erhält davon Eigenschaften, von denen einige in Metavariablen des Dokuments gespeichert werden:

- Verzeichnis (Meta-Variable `directory`)
- Template (Meta-Variable `template`)
- Metadatei (Meta-Variable `metafile`)
- Workflow-Einstiegspunkt

Weitere Informationen finden Sie in Kapitel 6 **Rubriken**

1.6 Workflow

Häufig haben sich in einem Unternehmen bereits eine Reihe von Arbeitsschritten etabliert, mit denen Inhalte ins Internet, Intranet und/oder Extranet gestellt werden. Hierzu gehören zum Beispiel Arbeitsschritte wie *beschreibende Informationen zu einer Seite eingeben, Inhalte in eine Seite eingeben und neue Seite überprüfen und freigeben oder ablehnen*.

In Imperia wird eine Reihe von Arbeitsschritten als Workflow bezeichnet. Imperia-Workflows sind völlig frei definierbar, man kann fast jede denkbare Reihe von Arbeitsschritten abbilden. Jedes Dokument muss seinen Workflow durchlaufen, bevor es fertig ist und freigeschaltet werden kann.

Jeder Schritt eines Workflows wird durch ein Workflow-Plug-In abgebildet. Im Lieferumfang sind eine Reihe von Workflow-Plug-Ins enthalten, die eine Vielzahl von Workflow-Funktionen zur Verfügung stellen. Durch das Aneinanderreihen und Verbinden der einzelnen Plug-Ins wird der Workflow gebildet. Als Tool zum Erzeugen und Verändern von Workflows steht eine grafische Oberfläche zur Verfügung - das so genannte Grid.

Man unterteilt Workflow-Plug-Ins in interaktive und nicht-interaktive Plug-Ins. Interaktive Plug-Ins erwarten an dieser Stelle des Workflows die Eingabe eines Users. Ein Beispiel hierfür ist das Bearbeiten-Plug-In. Dieses Plug-In stellt das Template in der Edit-Ansicht dar und erwartet, dass ein User Inhalt eingibt und seine Änderungen speichert.

Ein nicht-interaktives Plug-In ist beispielsweise das Metasetter-Plug-In, das eine Meta-Variable eines Dokuments auf einen eingestellten Wert setzt. Hierdurch können zum Beispiel auch Steuerungsvariablen für den Workflow definiert werden.

Neben Plug-Ins, die Meta-Variablen eines Dokuments manipulieren, gibt es auch Plug-Ins, die den Wert von Meta-Variablen überprüfen. Ein Beispiel hierfür ist das IfElse-Plug-In, das abhängig von dem Wert einer Meta-Variablen den weiteren Weg des Dokuments durch den Workflow ändern kann.

Man hat also sowohl die Möglichkeit, Meta-Variablen in einem Dokument zu setzen als auch die Möglichkeit, den Wert einer Meta-Variablen zu überprüfen und daraufhin den weiteren Workflow eines Dokuments zu bestimmen. Diese beiden Möglichkeiten machen den Workflow in Imperia zu einem sehr mächtigen und komfortablen Werkzeug, wenn es darum geht, den bereits im Unternehmen vorhandenen Workflow in Imperia nachzubilden.

Weitere Informationen finden Sie in Kapitel 8 **Workflows**

1.7 IWE - Imperia-WYSIWYG-Editor

Das IWE wird ab der Imperia Version 8.5.4 ausgeliefert. Um das IWE nutzen zu können, benötigen Sie einen neuen Imperia-Lizenzschlüssel.

Weitere Informationen finden Sie in Kapitel 5 **IWE - Imperia-WYSIWYG-Editor**

Kapitel 2. Installation von Imperia

Die Installation von Imperia kann unter folgenden Betriebssystemen erfolgen:

- UNIX® / UNIX® -Derivate (SuSE, RedHat, ...)
- Windows® 2000 / 2003 / XP

Die Installation wird mit Hilfe des interaktiven Installations-Skripts `install.pl` durchgeführt, mit dem sowohl Produktions- als auch Zielsysteme installiert werden können. Das Skript kann mit folgenden Optionen gestartet werden:

-h | -help

Zeigt die Liste der Optionen.

-l | -live *

Installiert ein Zielsystem. Lesen Sie hierzu auch Abschnitt 2.4 **Installation eines Zielsystems** auf Seite 11.

-c | -cgibin Pfad *

Absoluter Pfad zum CGI-BIN-Verzeichnis.

-s | -site Pfad *

Absoluter Pfad zum SITE-Verzeichnis.

-d | -docroot Pfad *

Absoluter Pfad zum Document-Root-Verzeichnis.

-w Location Pfad *

URL der cgi-bin-Location des Webservers.

-k | -keep

Überschreibt nie bereits vorhandene Dateien, sondern legt Backup-Kopien dieser Dateien an.

-r | -rm | -remove

Überschreibt immer bereits vorhandene Dateien und legt nie Backup-Kopien dieser Dateien an.

-f | -force

Erzwingt Neu-Installation der Konfigurations-Verzeichnisse und Dateien.

-n | -noprompt

Installation ohne Bestätigung der einzelnen Punkte durch den User.

-p

Bewahrt die Dateien `system.conf` und `imperia.conf` auch dann, wenn die dort gefundenen Pfade von denen über die Kommandozeile oder User-Eingabe übergebenen abweichen.

-rf | -fr

Gleichzeitig `-force` und `-remove`.

Optionen, die mit einem Stern gekennzeichnet sind, brauchen Sie nur zu verwenden, wenn Sie die Abfrage der entsprechenden Angaben während des Installationsprozesses überspringen wollen.

Alle vom Installationsskript ausgeführten Aktionen werden in einem Installationsprotokoll (`install.log`) protokolliert, das in dem Verzeichnis abgelegt wird, in dem das Installations-Skript gestartet wurde.

In diesem Protokoll werden alle Aktionen des Skripts und Ihre Eingaben gespeichert. Halten Sie diese Datei für eventuelle Supportanfragen bereit.

2.1 Voraussetzungen

Um Imperia installieren zu können, müssen für alle Betriebssysteme folgende Voraussetzungen geschaffen werden:

2.1.1 Dateisystem

Die Performance von Imperia hängt - insbesondere bei der Verwendung der Datenhaltung File65 (siehe Abschnitt 3.8 **Dokumentenhaltung** auf Seite 27) - stark vom verwendeten Dateisystem ab. Folgende Faktoren sollten berücksichtigt werden:

- Das Dateisystem sollte für die Verwaltung vieler, kleiner Dateien optimiert sein. Unter Linux muss daher von der Verwendung von ext3 (oder ext2) dringend abgeraten werden. Bewährt hat sich das Dateisystem ReiserFS.
- Die Verwendung von Netzwerkdateisystemen, insbesondere NFS wirkt sich negativ auf die Performance aus. Sollte NFS zum Einsatz kommen, sind Tuningmaßnahmen nötig.
- Der in `site/config/system.conf` konfigurierte Pfad für das Site-Verzeichnis sollte kein symbolischer Link sein.
- Weitere dateisystemspezifische Optimierungsmaßnahmen, zum Beispiel Defragmentierung, sollten regelmäßig durchgeführt werden.
- Die Leistungsfähigkeit aller Dateisystem leidet unter Betrieb an der Kapazitätsgrenze. Eine Belegung von mehr als 90 Prozent des verfügbaren Platzes sollte dringend vermieden werden.

2.1.2 Perl

Installieren Sie die Perl-Version 5.6.1., 5.8.1. oder eine höhere Version.



Achtung:

Die Perl-Versionen 5.6.0 von ActiveState und 5.8.0 enthalten Fehler bzw. Änderungen, die einen reibungslosen Betrieb von Imperia verhindern. Perl-Versionen ab 5.8.1 können hingegen zum Betrieb von Imperia verwendet werden.

Für Windows® -Betriebssysteme können Sie Perl unter folgenden Adressen herunterladen:

<http://www.activestate.com/>



Wichtig:

Verwenden Sie bitte unbedingt die Windows-Installer-Version des ActiveState Interpreters (.msi), da bei der Batch-Installation Probleme mit der Ausführung von CGI-Skripts auftreten sind.*

2.1.3 Perl-Modul Storable

Installieren Sie das Perl-Modul **Storable**. Sie können das Modul unter folgender Adresse für alle Betriebssysteme herunterladen:



Hinweis:

Wenn Sie die Perl-Version 5.8.X verwenden, ist die Installation des Storable-Moduls nicht nötig.

<http://www.perl.com/CPAN/modules/by-module/Storable/>

Wenn Sie Windows®-Betriebssysteme verwenden, können Sie das Modul **Storable** automatisch herunterladen und installieren. Gehen Sie dazu wie folgt vor:

1. Setzen Sie gegebenenfalls die Systemvariable HTTP_PROXY auf den für Ihr Netzwerk korrekten Wert. Fragen Sie im Zweifel Ihren Netzwerkadministrator. Das Verfahren zum Setzen von Systemvariablen entnehmen Sie bitte Ihrer Windows®-Dokumentation.
2. Öffnen Sie eine DOS-Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install Storable
```

Unter UNIX® können Sie das Modul **Storable** mit folgendem Befehl automatisch herunterladen und installieren (nur mit Root-Rechten):

```
perl -MCPAN -e 'CPAN::install Storable'
```

2.1.4 Perl-Modul DB_File

Für den Betrieb von Imperia benötigen Sie das Perl-Modul DB_File, das Sie vom CPAN herunterladen und installieren können.

http://www.perl.com/CPAN/modules/by-module/DB_File/

Für Windows bietet sich die Installation mittels ppm an.

2.1.5 Perl-Modul BerkeleyDB

Das Perl-Modul BerkeleyDB brauchen Sie ebenfalls für den Betrieb der Imperia Volltextsuche. Sie können es unter folgender URL kostenlos herunterladen:

<http://www.perl.com/CPAN/modules/by-module/BerkeleyDB/>

Für Windows bietet sich die Installation mittels ppm an.

2.1.6 Perl-Modul DBI

Damit Sie das Modul DBD::SQLite nutzen können, benötigen Sie das modul DBI. Mindestversion ist 1.21. Können das Modul unter folgender URL herunterladen:

<http://search.cpan.org/CPAN/authors/id/T/TI/TIMB/DBI-1.56.tar.gz>

Für Windows bietet sich die Installation mittels ppm an.

2.1.7 Perl-Modul DBD::SQLite

Das Perl-Modul DBD::SQLite brauchen Sie für die dateisystembasierte Datenhaltung File65. Sie können es unter folgender URL kostenlos herunterladen:

<http://search.cpan.org/CPAN/authors/id/M/MS/MSERGEANT/DBD-SQLite-1.13.tar.gz>

Für Windows bietet sich die Installation mittels ppm an. Allerdings ist die benötigte Version 1.13 für die Perl-Version 5.6.X nicht direkt über das ActiveState-Repository verfügbar. Wenn Sie diese Version einsetzen, können Sie das Modul unter folgender URL herunterladen:

<http://theoryx5.uwinnipeg.ca/ppmpackages/zips/DBD-SQLite.zip>



Wichtig:

Zum Betrieb dieses Moduls muss das Modul DBI mindestens in Version 1.21 vorliegen.

2.1.8 ImageMagick

Wenn Sie das Media Asset Management verwenden, installieren Sie das Modul **ImageMagick**. Dieses Modul können Sie unter folgender Adresse herunterladen:

<http://www.imagemagick.com/www/download.html>

Bei vielen Linux- und UNIX®-Distributionen ist ImageMagick Bestandteil der Distribution und kann wie jedes andere Paket über die entsprechenden mitgelieferten Tools installiert werden.



Achtung:

Bei der "Enterprise Edition" der SuSE AG ist ImageMagick nicht Bestandteil der Distribution, sondern muss nachinstalliert werden. Bei der Professional Edition fehlt die Schnittstelle zu Perl, PerlMagick.

2.1.9 Webserver

Installieren Sie einen CGI-fähigen Webserver. Wir empfehlen den Apache Webserver.

<http://httpd.apache.org>



Wichtig:

Bitte beachten Sie, dass der Internet Explorer keine Cookies akzeptiert, bei denen der Hostname Unterstriche "_" enthält (siehe auch Abschnitt 13.4.10.1 **Variable COOKIE-DOMAIN** auf Seite 215). Imperia benötigt Cookies für den Betrieb.

2.1.10 Zusätzliche Voraussetzungen für SQL-basierte Datenhaltung

Wenn Sie ein SQL-RDBMS für die Datenhaltung einsetzen möchten, müssen zusätzlich folgende Voraussetzungen erfüllt sein:

2.1.10.1 Perl-Module

- DBD::* - Das passende Modul für das eingesetzte RDBMS, Download über www.cpan.org, bzw. Installation via ppm.

2.1.10.2 Oracle®

- PL/SQL muss installiert bzw. aktiviert sein

2.1.10.3 MS-SQL®

- Mindestversion MS-SQL Server 2005®, da die Unterstützung von EXCEPT und INTERSECT benötigt wird.

2.1.10.4 MySQL

- Mindestversion 5.x



Wichtig:

Für diese Version von MySQL gibt es kein MySQLmax mehr.

2.1.05 PostgreSQL

- Mindestversion 8.1
- Pl/PGSQL muss installiert und aktiviert sein

2.1.11 Weitere Informationen

Halten Sie während der Installation folgende Informationen bereit:

- Document Root des Webservers
- IP-Adresse
- URL
- CGI-BIN-Verzeichnis
- User-ID des Webservers (nur UNIX®)

Bei gravierenden Installations-Problemen wenden Sie sich bitte unter folgender Email-Adresse an den Imperia-Support: <support@imperia.net>

2.2 Installation unter UNIX®-Derivaten

Zur Durchführung der Installation unter UNIX®-Derivaten gehen Sie wie folgt vor:

1. Achten Sie darauf, dass alle in Abschnitt 2.1 **Voraussetzungen** auf Seite 6 beschriebenen Voraussetzungen geschaffen wurden.
2. Falls Perl nicht in das Verzeichnis /usr/local/bin installiert wurde, legen Sie einen symbolischen Link von /usr/bin/perl nach /usr/local/bin/perl an.
3. Finden Sie heraus, unter welcher User-ID der Webserver läuft. Folgende Befehle liefern Hinweise:

```
cat /etc/passwd | grep -i www  
ypcat passwd | grep -i -E '(www|web)'
```

Überprüfen Sie gegebenenfalls auch die Konfigurationsdatei Ihres Webservers. Wenn Sie einen Apache-Webserver verwenden, suchen Sie in dessen Konfigurationsdatei nach den Begriffen `User` und `Group`:

```
egrep -i '(User|Group)' /path/to/httpd.conf
```

4. Nehmen Sie die in Schritt 3 gefundene User-ID an:

```
su wwwrun
```

5. Entpacken Sie das gelieferte Archiv in ein temporäres Verzeichnis.
6. Wechseln Sie in dieses Verzeichnis und starten Sie das Skript `install.pl`:

```
cd /Ihr/temporaeres/Verzeichnis  
../install.pl
```

Es startet die interaktive Installations-Routine.

Eine Übersicht über alle möglichen Installations-Optionen erhalten Sie am Anfang von Kapitel 1 dieses Buches oder über den folgenden Befehl:

```
perl install.pl -h
```

7. Wurde das Skript als **root** gestartet, müssen Sie zunächst die User-ID des Webservers angeben. Geben Sie dann an, ob Sie ein Produktionssystem (Develop) oder ein Zielsystem (Live) installieren wollen.
Geben Sie nacheinander die einzelnen Verzeichnisse an, die verwendet werden sollen. Hierzu gehören das CGI-BIN-Verzeichnis, das SITE-Verzeichnis und das Document-Root-Verzeichnis. Aus Sicherheitsgründen sollte das SITE-Verzeichnis außerhalb des Document-Root-Verzeichnisses liegen, um unerwünschte Zugriffe zu verhindern. Ausgehend vom Ursprungsverzeichnis Ihrer Installation werden nicht existierende Verzeichnisse angelegt.
Geben Sie auch den URL der `cgi-bin`-Location Ihres Webservers an.
8. Nachdem alle Fragen beantwortet wurden, erhalten Sie eine Übersicht über die getätigten Angaben. Überprüfen Sie diese und bestätigen Sie sie mit der Eingabe **yes**.
Möchten Sie einen Fehler korrigieren, geben Sie **no** ein und wiederholen Sie die Eingaben.
9. Nachdem Sie Ihre Eingaben bestätigt haben, werden die Dateien aus dem temporären Verzeichnis in die von Ihnen bestimmten Verzeichnisse kopiert.
10. Das temporäre Verzeichnis sollte erst nach der Installation eines Zielsystems gelöscht werden. Lesen Sie hierzu Abschnitt 2.4 **Installation eines Zielsystems** auf Seite 11.

2.3 Installation unter Windows® 2000 / 2003 Server / XP

Um Imperia unter Windows® 2000 / 2003 Server / XP zu installieren, gehen Sie wie folgt vor:

1. Achten Sie darauf, dass alle in Abschnitt 2.1 **Voraussetzungen** auf Seite 6 beschriebenen Voraussetzungen geschaffen wurden.
2. Entpacken Sie das gelieferte Archiv in ein temporäres Verzeichnis.
3. Starten Sie eine DOS-Eingabeaufforderung und wechseln Sie in das temporäre Verzeichnis.
4. Starten Sie das Installationsskript wie folgt:

```
perl install.pl
```

Eine Übersicht über alle möglichen Installationsoptionen erhalten Sie am Anfang dieses Buchs oder über den folgenden Befehl:

```
perl install.pl -h
```

5. Geben Sie an, ob Sie ein Produktionssystem (Develop) oder ein Zielsystem (Live) installieren wollen.
Geben Sie nacheinander die einzelnen Verzeichnisse an, die verwendet werden sollen. Hierzu gehören das CGI-BIN-Verzeichnis, das SITE-Verzeichnis und das Document-Root-Verzeichnis. Aus Sicherheitsgründen sollte das SITE-Verzeichnis außerhalb des Document-Root-Verzeichnisses liegen, um unerwünschte Zugriffe zu verhindern. Die Pfadangaben müssen mit Schrägstrichen (/) angegeben werden; andere Pfadangaben werden nicht verarbeitet. Verzeichnisse, die nicht existieren, werden, ausgehend vom Ursprungs-Verzeichnis, automatisch angelegt.
Geben Sie den URI der `cgi-bin`-Location Ihres Webservers an.
6. Nachdem alle Fragen beantwortet wurden, erhalten Sie eine Übersicht über die getätigten Angaben. Überprüfen Sie diese und bestätigen Sie sie mit der Eingabe **yes**.
Wenn Sie einen Fehler in Ihren Angaben korrigieren möchten, geben Sie **no** ein und korrigieren Sie die Eingaben.

7. Nachdem Sie Ihre Eingaben bestätigt haben, werden die Dateien aus dem temporären Verzeichnis in die von Ihnen bestimmten Verzeichnisse kopiert.
8. Das temporäre Verzeichnis sollte erst nach der Installation eines Zielsystems gelöscht werden. Lesen Sie hierzu Abschnitt 2.4 **Installation eines Zielsystems** auf Seite 11.

2.4 Installation eines Zielsystems

Um ein Zielsystem zu installieren, verfahren Sie genauso wie bei der Installation eines Produktionssystems. Lediglich bei der Auswahl des Servertyps (DEVELOP oder LIVE) geben Sie **LIVE** ein.

Im weiteren Verlauf der Installation müssen die entsprechenden Verzeichnisse für das Zielsystem angegeben werden. Verzeichnisse, die nicht existieren, legt das Skript, ausgehend vom Ursprungs-Verzeichnis, automatisch an.

2.5 Installieren eines Patches ("Hotfix")



Achtung:

Vor dem Einspielen eines Patches/Updates sollten Sie unbedingt ein Backup Ihres bestehenden Imperia-Systems durchführen.

Das Update eines laufenden Imperia-Systems wird besser durch das Einspielen eines Patches als durch eine komplette Neu-Installation durchgeführt.

Ein Imperia-Hotfix umfasst genau die Dateien, die für die neue Version geändert wurden. Hotfixe umfassen jedoch nur Änderungen bzw. Bugfixes bereits vorhandener Funktionen und Features. Neue Features werden nicht hinzugefügt.

Das Installationsskript überschreibt weder Daten noch vorhandene Konfigurationsdateien. Im Zweifelsfall werden vorhandene Dateien mit einer anderen Endung (.oo1, .oo2, .oo3, ...) gesichert.

Es werden jedoch ALLE Dateien überschrieben, die eine der folgenden Endungen haben:

.pl, .pm, .pod, .txt, .jar, .js, .de, .uk, .gif, .jpg, .jpeg, .tif, .tiff, .psd, .png, .pdf, .rtf, .doc, .xls, .bat, .sh, .exe

Wenn Sie in einer dieser Dateien Änderungen gemacht haben, besteht die Möglichkeit, diese Datei vom Skript sichern zu lassen. Starten Sie dazu das Installations-Skript mit der Option -k. Geänderte Dateien werden dann vor dem Kopieren umbenannt und im Installationsprotokoll aufgelistet.

Während des Updates werden vorhandene Dateien mit den neuen Dateien verglichen, wodurch erreicht wird, dass nach dem Update eine genau definierte Imperia-Version auf Ihrem System vorhanden ist. Das Installationsprotokoll sagt Ihnen genau, welche Dateien ersetzt wurden. Sichern Sie das Installationsprotokoll und halten Sie es für eventuelle Supportanfragen bereit.

Der Eintrag "Skipping" im Installationsprotokoll bzw. während der Installation zeigt, dass die betreffende Datei auf dem neuesten Stand ist und daher nicht ersetzt werden muss.

Um einen Patch bzw. ein Update zu installieren, gehen Sie wie folgt vor:

1. Führen Sie ein Backup Ihres bestehenden Imperia-Systems durch.
2. Entpacken Sie das Archiv, in dem das Update enthalten ist, in ein temporäres Verzeichnis.
3. Starten Sie in diesem Verzeichnis das Skript `install.pl`.
4. Wenn Sie den Patch unter UNIX® installieren, beachten Sie die Hinweise in Bezug auf die Zugriffsrechte für Dateien in Abschnitt 2.2 **Installation unter UNIX®-Derivaten** auf Seite 9.
5. Folgen Sie den Anweisungen des Skripts.

2.6 Update von einer älteren Imperia-Version



Achtung:

Vor dem Update sollten Sie unbedingt ein Backup Ihres bestehenden Imperia-Systems durchführen.

Wenn Sie von einer älteren Version von Imperia auf die aktuelle Version updaten möchten, gibt es ein paar Dinge, die Sie beachten müssen.

2.6.1 Referenzen

Die Erstellung und das Aktualisieren von Referenzen hat sich mit Imperia 8.6 geändert. Bitte führen Sie nach dem Update auf die aktuelle Version das Skript „site/bin/site_rebuild_db.pl“ aus:

```
site_rebuild_db.pl --full
```

Da dieser Vorgang einige Zeit in Anspruch nehmen kann, empfehlen wir diese Prozedur nachts durchzuführen. Für alle verfügbaren Parameter rufen Sie das Skript bitte mit dem Parameter „--help“ auf. Bitte denken Sie daran, vor dem Starten des Skripts den Hermes zu stoppen.

2.6.2 Update der Datenbankstruktur

Bitte führen Sie nach dem Update auf die aktuelle Imperia-Version ein Update der Datenbankstruktur durch. Weitere Informationen hierzu finden Sie im Kapitel Abschnitt 2.9 **Smart Database Update** auf Seite 15

2.7 Konfiguration von Imperia

Imperia wird mit Hilfe verschiedener Parameter in der Datei /site/config/system.conf konfiguriert. Im Folgenden finden Sie eine Liste der wichtigsten Variablen und deren Bedeutung. Weitere Konfigurationsvariablen sind im Anhang dieses Handbuchs unter Abschnitt 13.4 **Liste aller Variablen zur System-Konfiguration** auf Seite 209 aufgeführt.

Änderungen an den Dateien zur Systemkonfiguration können Sie über eine Schnittstelle in der Imperia-Benutzeroberfläche vornehmen. Lesen Sie hierzu Abschnitt 3.11 **Konfiguration des Systems über die Benutzeroberfläche** auf Seite 42.



Hinweis:

Sowohl der Parametername als auch der Wert müssen in Anführungszeichen gesetzt werden.

2.7.1 ALLOW_TEMPLATE_FORM

Erlaubt Formulare in Templates. Mögliche Einstellungen sind:

- 1: Das <form>-Tag kann in Templates verwendet werden.
- 0: Das <form>-Tag kann in Templates nicht verwendet werden.

2.7.2 ALLOW_TEMPLATE_SCRIPT

Erlaubt Plug-Ins und Skripte in Templates (z.B. JavaScript oder Applets). Mögliche Einstellungen sind:

- 1: Plug-Ins und Skripte können in Templates verwendet werden.
- 0: Plug-Ins und Skripte können nicht in Templates verwendet werden.

2.7.3 ALLOW_TEMPLATE_FRAME

Erlaubt Frames und Layer in Templates. Mögliche Einstellungen sind:

- 1: Frames und Layer sind erlaubt.
- 0: Frames und Layer sind nicht erlaubt.

2.7.4 UPLOAD_OBJECTS

Mit Hilfe dieses Parameters kann eingestellt werden, wie Imperia verlinkte Objekte beim Freischalten behandelt. Mögliche Einstellungen sind:

ALL

Es werden alle mit einem Dokument verlinkten Objekte freigeschaltet, die innerhalb der Attribute `src`, `background` oder `href` innerhalb der Tags `img`, `td`, `th`, `table`, `body`, `link`, `input` und `a` erscheinen.

LINKS

Diese Einstellung entspricht der Einstellung **ALL** mit dem Unterschied, dass hierbei das Anker-Tag (`<a>`) nicht berücksichtigt wird.

Kein Eintrag

Es werden alle mit einem Dokument verlinkten Objekte freigeschaltet, die innerhalb der Attribute `src` oder `background` innerhalb der Tags `img`, `td`, `th`, `table`, `body` erscheinen.

2.7.5 OPERATING-SYSTEM

Dieser Parameter beschreibt die Art der Plattform, auf der Imperia installiert ist. Mögliche Einstellungen sind:

- "DOS"
- "UNIX"

2.7.6 CGI-DIR

Dieser Parameter enthält den absoluten Pfad zum CGI-BIN-Verzeichnis des Produktionssystems, in dem alle CGI-Skripte von Imperia liegen.

2.7.7 DOCUMENT-ROOT

In diesem Parameter wird der absolute Pfad zum Document-Root-Verzeichnis des Webservers gespeichert. Das Verzeichnis `imperia` sollte ein Unterverzeichnis dieses Verzeichnisses sein (z.B.: `/usr/httpd/dev_serv/htdocs/imperia`).

2.7.8 CGI-BIN

In diesem Parameter wird der relative Pfad zum CGI-BIN-Verzeichnis des Webservers gespeichert.

2.7.9 DEMOMODE

Mit diesem Parameter wird der Demomodus ein- bzw. ausgeschaltet. Mögliche Einstellungen sind:

- 1: Demomodus ist aktiviert.
- 0: Demomodus ist nicht aktiviert.

2.7.10 SERVER-SCHEME

Dieser Parameter enthält das Protokoll-Schema, entweder HTTP oder HTTPS (HTTP mit SSL Verschlüsselung).



Hinweis:

Diese Variable müssen Sie nicht selbst setzen. Vielmehr liest Imperia die Einstellung aus der Umgebung des Webservers und speichert Sie in der Variablen ab.

2.7.11 SERVER_NAME

Dieser Parameter enthält den FQDN (Fully Qualified Domain Name) des Servers.



Hinweis:

Diese Variable wird automatisch gesetzt. (Siehe Abschnitt 2.7.10 SERVER-SCHEME auf Seite 13)

2.7.12 SERVER-PORT

Dieser Parameter enthält den Port, auf dem der Server läuft. Die Standardeinstellung ist 80 für HTTP und 443 für HTTPS (HTTP mit SSL Verschlüsselung).



Hinweis:

Diese Variable wird automatisch gesetzt. (Siehe Abschnitt 2.7.10 SERVER-SCHEME auf Seite 13)

2.7.13 HTTP_HOST

Dieser Parameter enthält das Verzeichnis, das via HTTP als Document Root angesprochen wird.

Standardeinstellung ist "/".

2.7.14 LOCK_EX, LOCK_SH, LOCK_UN, LOCK_NB

Zusätzliche Parameter zum File Locking.

2.7.15 COOKIE_TTL

Dieser Parameter enthält den Gültigkeitszeitraum (Time To Live) einer Usersession. Bleibt eine Usersession über diesen Zeitraum hinaus ungenutzt (idle), wird sie automatisch beendet und der User muss sich erneut anmelden.

2.7.16 SAFE_TEMPLATE

Wird dieser Parameter auf den Wert 1 gesetzt, dann wird Perl-Code im Template oder in Flexmodulen in einer sicheren Umgebung ausgeführt.

2.7.17 REG_KEY, REG_KEY2, REG_NAME

Diese Parameter enthalten die Lizenzschlüssel sowie den Kundennamen.

2.7.18 IMPERIA_ERROR_LOG

Mit diesem Parameter lenken Sie die Ausgabe auf STDERR in eine Datei um. Diese Datei befindet sich im Verzeichnis site/logs Ihrer Imperia-Installation und heißt error.log. Insbesondere bei der Verwendung eines Apache Webservers unter Windows® 2000 kann es zu Abstürzen kommen, wenn mehr als 1KB Daten nach STDERR beschrieben werden. Wird STDERR in eine Datei umgeleitet, werden diese Abstürze vermieden. Mögliche Werte sind 1 bzw. eine Zeichenkette für die Aktivierung der Umlenkung oder 0 für deren Deaktivierung.

2.7.19 LATCHING

Normalerweise verlassen Sie den Bearbeiten-Schritt eines Dokuments beim Abspeichern Ihrer Eingaben automatisch. Das System führt Sie dann auf den Schreibtisch. Sollten Sie eine Möglichkeit zum Speichern Ihrer Eingaben wünschen, ohne den Bearbeiten-Schritt verlassen zu müssen, können Sie diese mit der Variablen LATCHING aktivieren. Wenn Sie diese auf 1 setzen, erscheint im Bearbeiten-Schritt der Button **Zwischenspeichern**, mit dem Sie Ihre Eingaben abspeichern und das Dokument weiterhin bearbeiten können.

2.8 Modul-Installation

2.8.1 Verwendung des Grids

Beim Grid handelt es sich um die Oberfläche zum Erstellen und Ändern von Workflows. Die client-seitigen Voraussetzungen für die Verwendung des Grids sind:

- ein JavaScript-fähiger Browser, der außerdem den Tag `embed` oder `object` darstellen kann (Internet Explorer ab Version 4 (oder höher), Netscape Navigator ab Version 4 (oder höher)).
- Das JavaRuntime-Plug-In in der Version 1.3.1 (oder höher); erhältlich für die einzelnen Plattformen unter <http://java.sun.com/>.

2.8.2 Hinweise für das JavaRuntime-Environment



Achtung:

Die Version 1.3.0 des JavaRuntime Plug-Ins ist instabil! Achten Sie unbedingt darauf, die Version 1.3.1 (oder höher) auf den Client-Rechnern zu installieren, die mit dem Grid arbeiten sollen.

Es müssen keine JavaPolicy-Einstellungen für das Applet gemacht werden.

Windows®

Sollte bereits eine ältere JavaRuntime-Version auf einem Windows®-Client-Rechner installiert sein, genügt es, die neue Version darüber zu installieren.

Linux

Kopieren Sie die Datei `javaplugin.so` (ab Netscape 6.x `libjavaplugin_oji.so`) aus der Java-Runtime in das Netscape-Plug-In-Verzeichnis. Der System-Administrator kann diese Datei ins globale Plug-In-Verzeichnis kopieren (unter SuSE-Linux: `/opt/netscape/plugins`). Alternativ muss jeder User die Datei in sein Home-Verzeichnis unter `${HOME} / .netscape/plugins` ablegen.

2.9 Smart Database Update



Achtung:

Vor dem Update sollten Sie unbedingt ein Backup Ihres bestehenden Imperia-Systems durchführen.

Ab der Version 8.5.4, verwendet die SQL-Datenhaltung in Imperia mehr Constraints um die Datenbankintegrität besser sicherstellen zu können. Die Indexe im Datenbankschema wurden überarbeitet um mehr Leistung zu erzielen. Diese Indexe werden nicht automatisch aktualisiert. Sehen Sie hierfür bitte im Kapitel 'Upgrade und Wartung' nach. Vor Imperia 8.5.4 war das Updaten des Datenbankschemas ein mühseliger Prozess. Die Dokumentenhaltung musste zuerst nach File65 umgewandelt werden. Anschliessend wurde eine leere Datenhaltung mit dem neuen Datenbankschema erstellt, welche zu guter Letzt wieder mit den vorher kopierten Daten gefüllt werden musste. Da dieser Vorgang sehr Zeitintensiv war, konnte es zu beachtlichen Ausfallzeiten kommen. Zusätzlich verursachte der gesamte Prozess eine erhebliche Auslastung. Mit der Zeit verbesserte der 'batch-mode' des `clone_repository.pl` Skripts diese Situation ein wenig, doch war eine erforderliche Ausfallzeit unumgänglich.

2.9.1 Update und Wartung

Seit Imperia 8.5.4 werden zwei neue Skripte mitgeliefert, welche das Upgraden und die Wartung der Datenhaltung erheblich erleichtern.

2.9.1.1 site_doctree_admin.pl

In Imperia 8.5.4 wurde das Skript `site/bin/site_doctree_admin.pl` hinzugefügt, welches die Datenbankmigration in Echtzeit durchführt. Wann immer es möglich ist, teilt das Skript den Vorgang in kleine Einzelvorgänge auf. So ist es möglich den Vorgang zu unterbrechen und später an der gleichen Stelle fortzusetzen.



Hinweis:

Bitte beachten Sie, dass nur die Datenhaltungen PostgreSQL und SQLite ein Rollback für Änderungen in der Datenbankstruktur (Tabellen hinzufügen/löschen, Constraints, etc.) unterstützen.

Für die Datenhaltungen Oracle und SQLite gibt es ein paar wenige Bedingungen. Somit kann eine unterbrochenen Migration evtl. den manuellen Eingriff in den Prozess erfordern.

Wenn Sie das Skript ohne jegliche Parameter aufrufen, analysiert dieses die aktuelle Version des Datenbankschemas und schlägt evtl. ein Update vor.

Mit dem Aufruf:

```
site_doctree_admin.pl --init
```

starten Sie die Migration der Datenbank. Dieser Aufruf kann auch mit der Option „--dry-run“ kombiniert werden. Das führt dazu, dass keine wirklichen Änderungen durchgeführt werden, sondern lediglich eine Migration simuliert wird.

Alle verfügbaren Parameter erhalten Sie mit:

```
site_doctree_admin.pl --help
```

2.9.1.2 site_doctree_repair.pl

Wenn Sie die Änderungen durchführen kann es in einzelnen Fällen zu Problemen mit doppelt vorhandenen Schlüsseln kommen. Für diesen Fall steht das Skript `site_doctree_repair.pl` zur Verfügung, welches versucht die doppelten Zeilen aus der Datenbank zu entfernen. Wenn der Aufruf von

```
site_doctree_admin.pl --init
```

fehlschlägt, können Sie in der Ausgabe des Skripts nachsehen, welche Tabelle den Fehler verursacht hat. Wenn Sie erst einmal die betroffene Tabelle kennen, können Sie diese mit `site_doctree_repair.pl` versuchen zu reparieren:

```
site_doctree_repair.pl --table=TABLE --repair
```

Auch diesen Aufruf können Sie zuerst mit dem Parameter `--dry-run` kombinieren um zu sehen, was das Skript für Änderungen an der jeweiligen Tabelle vornehmen wird. Alle verfügbaren Parameter können Sie mit

```
site_doctree_repair.pl --help
```

nachsehen.

Kapitel 3. Administration des Systems

Alle für die Systemadministration relevanten Menüs und Dialoge werden über den Ordner **System** geöffnet.

3.1 Erstmaliges Starten und Anmelden

Um Imperia zum ersten Mal zu starten, geben Sie in der Adressleiste Ihres Browsers folgenden Link ein:

`http://<Ihr Produktions-Server>/imperia/`

Es wird die Login-Maske geöffnet:



Abb. 3.1: Login-Maske

Zum erstmaligen Anmelden an Imperia wurde bei der Installation der User **superuser** angelegt. Als Passwort wurde für diesen User ebenfalls **superuser** eingerichtet. Der Superuser hat Zugriff auf alle Funktionen und Menüs, konfiguriert und pflegt das System und legt User und Rollen an.



Achtung:

Der Superuser hat alle Berechtigungen im System. Arbeiten Sie unter diesem User nur, wenn Sie das entsprechende Wissen über die einzelnen Menüpunkte haben!

Starten Sie Imperia und tragen Sie in das Feld **Name** der Login-Maske **superuser** ein. In das Feld **Password** tragen Sie ebenfalls **superuser** ein. Klicken Sie anschließend auf den Button **Okay**. Daraufhin wird das Hauptmenü geöffnet.

Vervollständigen Sie nun die Konfiguration von Imperia. Führen Sie dazu folgende Schritte durch:

1. Ändern Sie das voreingestellte Passwort für den Superuser.
2. Installieren Sie ein Zielsystem, falls dies noch nicht geschehen ist. Lesen Sie hierzu Abschnitt 2.4 **Installation eines Zielsystems** auf Seite 11.
3. Tragen Sie die Lizenzdaten ein, damit alle Funktionen freigeschaltet werden. Lesen Sie hierzu Abschnitt 3.3 **Lizenzierung** auf Seite 19.
4. Geben Sie die Daten des Standard-Zielsystems ein. Lesen Sie hierzu Abschnitt 3.4 **Standard-Zielsystem** auf Seite 19.

3.2 Wartungsmodi für Imperia

Für einige administrative Aufgaben ist es sinnvoll, vor der Durchführung dieser Wartungsarbeiten das Produktionssystem in einen Wartungsmodus zu versetzen. Im Wartungsmodus können die User nicht auf das System zugreifen, ohne dass der Webserver gestoppt werden muss.

In Imperia stehen zwei unterschiedliche Wartungsmodi zu Verfügung, die den Zugriff auf das System unterschiedlich stark einschränken.

3.2.1 Wartungsmodus ohne Superuser-Zugriff

Sobald dieser Wartungsmodus aktiviert wird, werden die Klicks aller User (auch des Superusers) in die Login-Maske umgelenkt, wo sie sich erst wieder anmelden können, wenn der Wartungsmodus wieder abgeschaltet wurde.



Abb. 3.2: Wartungsmodus

Um das Produktionssystem in diesen Wartungsmodus zu versetzen, setzen Sie folgende Variable in der Datei `site/config/system.conf`:

```
"AUTH_PLUGIN" = "MaintenanceNoLogin"
```



Hinweis:

Sie müssen diese Variable von Hand in der `site/config/system.conf` nachtragen, da sie bei der Installation nicht automatisch dort angelegt wird.

Um den Wartungsmodus wieder zu verlassen, müssen Sie den Wert der `system.conf`-Variablen wieder auf den Standard-Wert `Password` setzen oder die entsprechende Zeile auskommentieren (mit vorangestellter Raute #) bzw. löschen.

Dieser Modus bietet sich beispielsweise an, wenn Sie unter einer SQL-Datenhaltung ein Backup Ihrer Datenbank erstellen wollen und verhindern möchten, dass während des Backup-Prozesses durchgeführte Änderungen am Content verloren gehen.

3.2.2 Wartungsmodus mit Superuser-Zugriff

Während der erste Wartungsmodus alle Log-In-Versuche unterbindet, kann sich in diesem Modus der Superuser am System anmelden. Das kann bei bestimmten Wartungsarbeiten erforderlich sein.

Die Aktivierung können Sie entweder im Dialog *Allgemeine Einstellungen* vornehmen (siehe Abschnitt 3.10.6 **Superuser Authentifizierungs-Plug-In** auf Seite 41) oder indem Sie die Datei `system.conf` editieren:

```
"AUTH_PLUGIN" = "Superuser"
```

Um das System wieder in den Normalbetrieb zu versetzen, verfahren Sie wie beim Wartungsmodus ohne Superuser-Zugriff bzw. deaktivieren Sie die entsprechende Option im Dialog *Allgemeine Einstellungen*.

3.3 Lizenzierung

Die Lizenzierung schaltet die erworbenen Funktionen in Imperia frei. Mit der Software erhalten Sie einen Registrierungsschlüssel und den Namen des Lizenznehmers. Beide werden in den Dialog Registrierungsschlüssel eingetragen. Achten Sie hierbei auf Groß- und Kleinschreibung oder Leerstellen.

Falls in Zukunft weitere Funktionen oder Module erworben werden, müssen die alten Daten durch die neuen Daten ersetzt werden.

Zur Registrierung von Imperia gehören:

- die Datei `regkey.bin`
- der Name des Lizenznehmers
- ein 24-stelliger Registrierungsschlüssel

Kopieren Sie zunächst die Datei `regkey.bin` in das Verzeichnis `site/config` auf dem Produktionssystem. Die Datei finden Sie auf Ihrer CD. Starten Sie anschließend Imperia und melden Sie sich als User mit entsprechenden Rechten an, im Zweifel als Superuser. Öffnen Sie den Dialog **Registrierungsschlüssel** über den Link **Registrierungsschlüssel** des Ordners **System**. Sie gelangen daraufhin direkt in folgenden Eingabedialog:

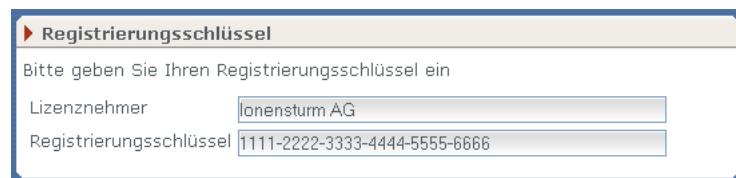


Abb. 3.3: Dialog Registrierungsschlüssel

Tragen Sie den Lizenznehmer und den Registrierungsschlüssel in die gleichlautenden Felder des Dialogs ein. Achten Sie unbedingt auf Groß- und Kleinschreibung und auf die exakte Übernahme der Schreibweise des Lizenznehmers. Klicken Sie anschließend auf den Button **Okay** am unteren Rand der Imperia-Oberfläche. Der Button **Abbrechen** bricht die Lizenzierung ab, ohne Eingaben zu speichern.

3.4 Standard-Zielsystem

Um mit Imperia Dokumente freischalten zu können, müssen die Daten mindestens eines Zielsystems bekannt sein. Diese Daten werden in den Dialog **Standard-Zielsystem** eingetragen. Der Dialog wird über den Menüpunkt **Standard-Zielsystem** des Ordners **System** geöffnet.

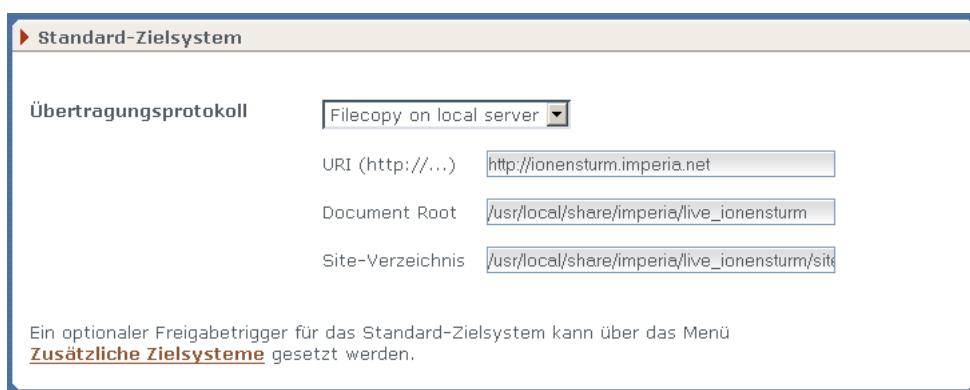


Abb. 3.4: Dialog Standard-Zielsystem

Der Dialog besteht aus einer Drop-Down-Box und verschiedenen Eingabefeldern.

Über die Drop-Down-Box wird das Protokoll eingestellt, mit dem fertige Dokumente auf das Zielsystem übertragen werden sollen. Folgende Protokolle stehen zur Auswahl:

- Filecopy on local server
- FTP-Transfer

- SFTP-Transfer
- HTTP-Transfer
- Secure Copy via SSH
- WebDAV (mit und ohne SSL)

Zu jedem Protokoll werden das zu verwendende Protokoll, mit dem Dateien auf das Zielsystem übertragen werden (z.B. http, https, ftp), das Document-Root und das Site-Verzeichnis des Zielsystems in die entsprechenden Felder eingegeben. Alle Felder müssen ausgefüllt werden. Folgende Felder sind möglich:

Document-Root

Hier muss das Standard-Verzeichnis für HTML-Dateien eingetragen werden. Oft ist dies /var/htdocs oder c:/inetpub/wwwroot.

SITE-Verzeichnis

Beim Site-Verzeichnis handelt es sich um ein Standard-Verzeichnis für Imperia; es stellt die Basis für alle Imperia-Module dar. Beispiel: /pfad/zur/imperia/installation/site

Die Verzeichnisse, die Sie unter Document-Root und Site-Verzeichnis eintragen, müssen bereits existieren. Achten Sie darauf, Verzeichnisangaben grundsätzlich ohne schließenden Schrägstrich (Slash: /) zu tätigen. Auch unter Windows® müssen Verzeichnisse mit Hilfe des Schrägstrichs und nicht mit Backslashes (\) angegeben werden.

Weitere in diesem Dialog mögliche Felder sind:

URI (<http://...>)

Tragen Sie hier den URI für Ihr Zielsystem ein (z.B.: <http://www.ihredomain.de>).

IP-Adresse

Tragen Sie hier die IP-Adresse des Zielsystems ein, wenn Dokumente mit FTP oder HTTP freigeschaltet werden. Optional können Sie hier auch den Benutzernamen ergänzen, beispielsweise bei der Verwendung von SecureCopy (wwwrun@meinserver.de).

Login

Geben Sie hier die Anmeldekennung an, falls Ihr Zielsystem dies erfordert.

Passwort

Tragen Sie hier das Passwort ein, falls das Zielsystem dies erfordert.

Passiver FTP-Modus

Wird diese Checkbox markiert, verwendet Imperia den passiven FTP-Modus, um Dokumente per FTP auf das Standard-Zielsystem zu übertragen.

Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Angaben zu speichern. Die aktuellen Parameter werden anschließend zusammengefasst dargestellt. Der Button **Abbruch** bricht die Aktion ab und Sie gelangen zurück in das Hauptmenü, ohne dass Eingaben gespeichert werden.

3.5 Systemdienste

Imperia startet Systemdienste zu bestimmten Zeitpunkten automatisch, um SiteActives zur Erzeugung von Übersichtsseiten oder Datenbank-Exportaufträge auszuführen. Voraussetzung für die Nutzung von Systemdiensten ist ein laufender Hermes-Daemon auf dem Entwicklungssystem und allen Zielsystemen, auf denen Sie den Systemdienst nutzen wollen. Lesen Sie hierzu auch den Abschnitt Kapitel 11 **Der Daemon Hermes**.

Die Konfiguration der auf Ihrem Imperia-System laufenden Dienste nehmen Sie im Dialog **Hauptmenü->System->Systemdienste** vor. Imperia speichert die Systemdienstkonfiguration in der Datei `site/config/imperia.schedule` Ihres Entwicklungssystems. Diese Datei muss auch auf allen Zielsystemen vorliegen, auf denen die Systemdienste laufen sollen. Zur Übertragung dieser Konfigurationsdatei nutzen Sie die Funktion **Hauptmenü->System->Systemdateien abgleichen**.

Den Zeitpunkt für die Ausführung können Sie entweder durch zeitlich- oder ereignisgesteuert definieren. Für zeitlich gesteuerte Systemdienste sind folgende Optionen für die Ausführung verfügbar:

- in bestimmten Intervallen (stündlich, täglich, nach Ablauf bestimmter Minuten- oder Stundenfristen)
- zu bestimmten Uhrzeiten
- an bestimmten Wochentagen
- monatlich

Ereignisgesteuerte Systemdienste richten Sie ein, indem Sie eines oder mehrere Verzeichnisse angeben, die der Hermes-Daemon überwachen soll. Wenn Sie in einem überwachten Verzeichnis ein neues Dokument hinzufügen oder ein bestehendes löschen oder verändern, führt der Hermes-Daemon den entsprechenden Systemdienst aus.



Hinweis:

Beide Varianten sind voneinander unabhängig. Sie können also beispielsweise ein SiteActive sowohl durch einen zeitlich gesteuerten Systemdienst als auch bei Änderungen in einem bestimmten Verzeichnis ausführen lassen. Achten Sie bei der Kombination von ereignis- und zeitlich gesteuerten Systemdiensten darauf, dass der Hermes-Daemon das ausgeführte SiteActive nicht zu selten oder zu häufig ausführt.

Wenn Sie für ein SiteActive beispielsweise einen ereignisgesteuerten und einen zeitlich gesteuerten Systemdienst mit einem langen Intervall (z.B. monatlich) definiert haben, geschieht die Ausführung fast ausschließlich durch Änderungen im überwachten Verzeichnis. Ohne den ereignisgesteuerten Systemdienst startet nur der zeitgesteuerte Dienst das SiteActive. Bei großen Zeitintervallen besteht dann beispielsweise die Gefahr, dass gelöschte Artikel vorübergehend zu toten Links führen.

Treten hingegen in einem überwachten Verzeichnis häufig Änderungen ein und ist zusätzlich ein Systemdienst mit einem kurzen Zeitintervall von beispielsweise fünf Minuten definiert, kann es passieren, dass der Hermes-Daemon das betreffende SiteActive mehrfach innerhalb eines Durchlaufs ausführt.

Der Hermes-Daemon entnimmt der Konfiguration des Systemdienstes, welches Template für die Ausführung des SiteActives dient und wo die daraus generierte Datei unter welchem Namen abzuspeichern ist. Diese Parameter über gibt er einem CGI-Skript, dass Sie zum Testen des SiteActive-Codes auch manuell aufrufen können. Lesen Sie hierzu den Abschnitt "**SiteActives manuell aufrufen**" im Kapitel "**SiteActive**" des Programmierhandbuchs.



Achtung:

Die Ausführung eines Systemdienstes in kurzen Intervallen oder mehrerer Systemdienste gleichzeitig kann zu erheblicher Systemlast auf dem Server führen.

3.5.1 Neuen Systemdienst anlegen

Rufen Sie den Dialog für die Systemdienstkonfiguration auf: **Hauptmenü->System->Systemdienste**.



Abb. 3.5: Dialog zum Anlegen eines Systemdienstes

Klicken Sie in einem der Abschnitte auf den Link **Neuer Eintrag**. Es erscheint folgender Dialog:

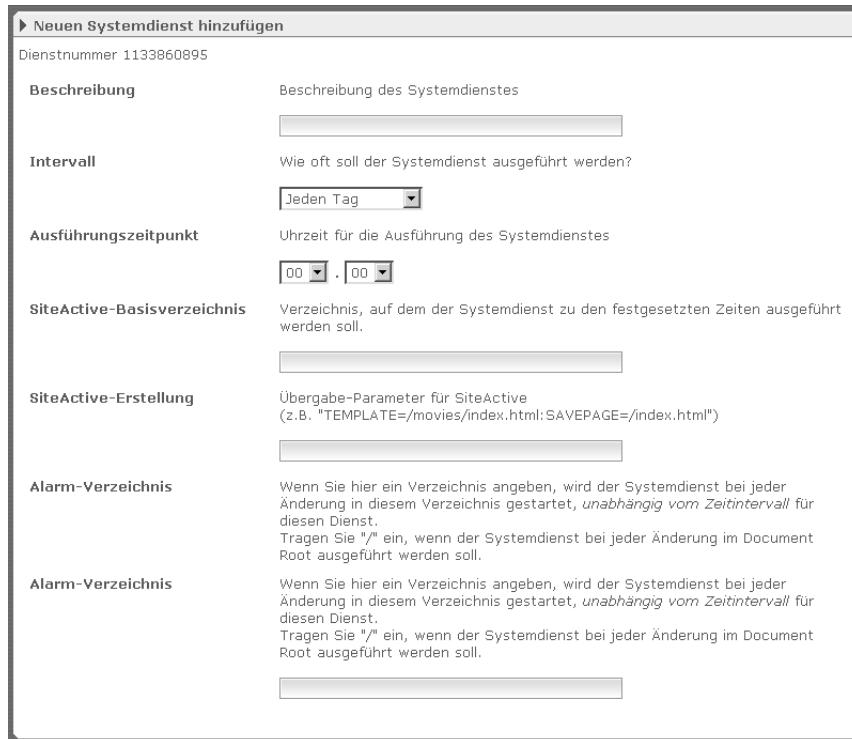


Abb. 3.6: Dialog Neuen Systemdienst anlegen

Die Zuordnung eines Dienstes zu einem Abschnitt geschieht erst, wenn Sie einen Zeitpunkt für die Ausführung des Dienstes angeben. Wenn Sie also den Eingabedialog für einen neuen Systemdienst über den Abschnitt **Wöchentliche Dienste** öffnen, den Dienst jedoch viertelstündlich ausführen lassen, erscheint der Diensteintrag nach dem Speichern im Abschnitt **Tägliche Dienste**.

Jeder Systemdienst erhält automatisch eine Dienstnummer, die Sie im oberen Teil des Dialogs sehen. Das Feld **Beschreibung** dient für eine kurze Beschreibung der Funktion des Dienstes, die in der Liste der Systemdienste als Dienstüberschrift erscheint. Mit der Drop-Down-Box **Intervall** bestimmen Sie bei zeitlich gesteuerten Systemdiensten das Ausführungsintervall. Die beiden Drop-Down-Boxen unter der Überschrift **Ausführungszeitpunkt** dienen zum Einstellen einer bestimmten Uhrzeit. Mit der linken Drop-Down-Box stellen Sie die Stunden und mit der rechten die Minuten im Viertelstundentakt ein.

In das Feld **SiteActive-Basisverzeichnis** tragen Sie das Verzeichnis ein, in dem der Systemdienst zu den festgesetzten Zeiten auszuführen ist.

Im Abschnitt **SiteActive-Erstellung** tragen Sie die Parameter ein, die für die Ausführung des SiteActive notwendig sind. Trennen Sie dabei die einzelnen Parameter durch Doppelpunkte voneinander. Pflichtparameter sind:

- Angabe des Templates: TEMPLATE=/ver/zeich/nis/Datei.name
- Zielseite: SAVEPAGE=/ver/zeich/nis/datei.html

Optional besteht die Möglichkeit, weitere Variablen in Form eines Name-Wert-Paars zu definieren. Diese Variablen lassen sich bei der Verarbeitung des SiteActive-Codes als FF-Variablen referenzieren.

Beispiel für einen Eintrag im Feld **SiteActive-Erstellung**:

```
TEMPLATE=/movies/index.html:SAVEPAGE=/index.html:foo=bar:myvar=myvalue
```

Im Feld **Alarm-Verzeichnis** tragen Sie das Verzeichnis ein, das der Hermes-Daemon für diesen Systemdienst überwachen soll.

Wenn Sie mehrere Verzeichnisse überwachen wollen, tragen Sie diese als kommaseparierte Liste in das Feld ein:

```
/verz1,/verz2,/verz3,/verz4/news/
```

Sie benötigen also nur einen Systemdienst für die Überwachung mehrerer Verzeichnisse.



Wichtig:

Die Pfadangaben für TEMPLATE, SAVEPAGE und Alarmverzeichnis interpretiert das System immer ausgehend von der Document-Root Ihres Imperia-Systems.

Wenn Sie bestimmte Verzeichnisse oder Dateien explizit von der Überwachung ausschließen möchten, tragen Sie diese mit vorangestelltem Ausrufezeichen in das Feld ein:

```
/verz1,!/verz1/print.html,/verz2/news,!/verz2/news/archive
```

Der Auslöser für den Systemdienst sind in diesem Fall die Notification Files , die im Verzeichnis /site/incoming eingehen. Der Daemon Hermes überwacht dieses Verzeichnis und prüft anhand der Pfadangaben, die in den Notifications stehen, ob ein dazu passender Systemdienst existiert. Stimmt die Pfadangabe mit dem Alarm-Verzeichnis eines Systemdienstes überein, stößt er dessen Ausführung an und löscht anschließend das Notification File. Lesen Sie hierzu auch den Abschnitt Abschnitt 3.7.1, „Notification File“.

Klicken Sie auf **Okay**, um den neuen Systemdienst zu speichern, oder auf **Schließen**, um den Dialog zu verlassen, ohne die Einstellungen zu speichern.

Führen Sie die Funktion **Hauptmenü->System->Systemdateien abgleichen** aus, um die geänderten Einstellungen auf die betreffenden Zielsysteme zu übertragen.

3.5.2 Systemdienst ändern

Um einen vorhandenen Systemdienst zu ändern, gehen Sie wie folgt vor:

1. Rufen Sie den Dialog für die Systemdienstkonfiguration auf: **Hauptmenü->System->Systemdienste**.

Alle bereits eingerichteten Systemdienste sind dort durch einen horizontalen Balken voneinander getrennt in mehreren Abschnitten aufgelistet.

Für jeden Abschnitt sind die angezeigten Systemdienste individuell sortierbar. Dafür stehen mehrere Sortieroptionen zur Verfügung, die Sie jeweils mit einer Pull-Down-Liste auswählen.

2. Klicken Sie im Abschnitt des zu ändernden Dienstes auf den Link **[edit]**. Es öffnet sich der Dialog **Systemdienst bearbeiten**. Die Einstellungsmöglichkeiten sind die selben wie beim Anlegen eines neuen Systemdiensts
3. Nehmen Sie die gewünschten Änderungen vor.
4. Klicken Sie auf **Okay**, um Ihre Änderungen zu speichern, oder auf **Schließen**, um sie zu verwerfen.
5. Sie gelangen zurück in den Dialog **Systemdienste**, wo Sie Ihre gespeicherten Änderungen kontrollieren können.

Die Änderung eines Systemdienstes ist damit abgeschlossen.



Wichtig:

Neue Systemdienste und Änderungen an einem bestehenden Systemdienst werden erst nach einem Abgleich der Systemdateien aktiv. Führen Sie dazu die Funktion Hauptmenü->System->Systemdateien abgleichen aus.

3.5.3 Systemdienst löschen



Achtung:

Beim Löschen von Systemdiensten erscheint keine Sicherheitsabfrage.

Um einen vorhandenen Systemdienst zu löschen, gehen Sie wie folgt vor:

1. Rufen Sie den Dialog für die Systemdienstkonfiguration auf: **Hauptmenü->System->Systemdienste**. In den einzelnen Abschnitten sind alle vorhandenen Systemdienste aufgelistet.
2. Klicken Sie auf den Link **[delete]** im Abschnitt des zu löschen Dienstes.



Wichtig:

Um den Systemdienst auch auf den Zielsystemen zu löschen, führen Sie die Funktion Hauptmenü->System->Systemdateien abgleichen aus.

3.6 Modul-Lizenzierung

In diesem Abschnitt wird auf die Lizenzierung des ActiveX-Wordmoduls und des EWE - Easy Web Editors sowie der uCrop-Bildverarbeitung eingegangen. Diese Module sind nicht im Standard-Lieferumfang von Imperia enthalten und müssen separat lizenziert werden.

Der Dialog **Modul-Lizenzierung** enthält drei relevante Abschnitte:

- Active-X-Wordmodul
- EWE - Easy Web Editor
- uCrop-Bildverarbeitung

Der Abschnitt **Konfiguration** bietet die Möglichkeit, schnell zu den anderen Abschnitte zu navigieren.

Im Abschnitt **EWE - Easy Web Editor** wird die Lizenzierung des EWE-Wordmoduls eingetragen. Tragen Sie Ihre Lizenzdaten in das Eingabefeld ein.

Der Abschnitt **ActiveX-Wordmodul** dient der Lizenzierung des ActiveX-Wordmoduls. Auch hier müssen die Lizenzdaten, die Sie mit der Lieferung erhalten haben, in die entsprechenden Felder eingetragen werden.

Im untersten Abschnitt wird der Registrierungsschlüssel des uCrop-Bildverarbeitungsmoduls eingegeben. Lesen Sie dazu die in diesem Modul beiliegende Dokumentation.

3.7 Zusätzliche Zielsysteme

Nach der Installation ist Imperia so eingestellt, dass freigeschaltete Dokumente und die dazugehörigen Medien immer auf das Standard-Zielsystem freigeschaltet werden. Im vorliegenden Abschnitt beschäftigen wir uns mit der Einrichtung sogenannter Freischalt-Trigger, die weitere Ziele für freigeschaltete Dokumente definieren und anhand derer Imperia das Zielsystem bestimmt, auf das ein freigeschaltetes Dokumente übertragen werden soll.

Ein Freischalt-Trigger besteht aus:

- der Beschreibung des Ereignisses (Triggers), das die Übertragung der Dateien auf einen Server auslöst.
- dem Zielformat der freigeschalteten Dateien.
- dem Namen und dem Pfad der Zielformat.
- der Angabe des Servers, auf den die Datei übertragen werden soll. Dies kann das Standard-Zielsystem, das Produktionssystem oder ein alternatives Zielsystem sein.

Weitere Informationen zu den Bestandteilen eines Freischalt-Triggers finden Sie im folgenden Abschnitt:

3.7.1 Neuen Freischalt-Trigger eintragen

Der Konfigurationsdialog für neue Freischalt-Trigger wird über den Link **Zusätzliche Zielsysteme** im Ordner **System** geöffnet. Es öffnet sich folgender Dialog:

Server	Übertragungsprotokoll	Zielformat	Dateipfad auf Zielsystem	Freischalt-Trigger	Schlüsselwert	Bearbeiten	Löschen
http://ionensturm.imperia.net STANDARD	Filecopy	HTML	docroot/dir/file.html	Always			
http://ionensturm.imperia.net STANDARD	Filecopy	Notification	site/incoming/time-count.meta	Always			
http://netweaver...net:50000/irj/go/km/docs/ XML	WebDAV	SAP_EP	docroot/dir/file.html	Metafield	filename ^sap_		
		XML	docroot/dir/file.html.xml	Metafield	target_xml ja		

Abb. 3.7: Dialog Zusätzliche Zielsysteme

Der Dialog zeigt die bereits eingerichteten Zielsysteme. Um ein neues zusätzliches Zielsystem einzurichten, klicken Sie auf den Button **Neues Zielsystem**. Es öffnet sich folgender Dialog:

The dialog box has the following fields:

- Zielformat:** HTML-Export
- Übertragungsprotokoll:** FTP-Transfer
- Zieldatei:** DOCROOT
- Freischalt-Trigger:** Immer

Abb. 3.8: Neues Zielsystem hinzufügen

In der Drop-Down-Box **Zielformat** wird das Format eingestellt, in dem Dokumente auf dem entsprechenden Zielsystem abgelegt werden sollen. Es stehen folgende Einstellungen zur Verfügung:

HTML

Das Dokument wird in dem Format, in dem es nach dem Verlassen des Workflows im Document-Root des Produktionssystems abgelegt wurde, übertragen und gespeichert. Somit gibt letztendlich das Format des zuletzt verwendeten Templates das Format des Dokumentes an. Die Werte der Variablen `directory` und `filename` geben den Speicherort des Dokumentes im Filesystem des Zielservers an.

Notification File

Es wird eine Datei erzeugt, die den kompletten Metadatenhash des freigeschalteten Dokumentes enthält. Diese Datei wird im Verzeichnis `/site/incoming` abgelegt und triggert über den Hintergrund-Dienst `Hermes`, der dieses Verzeichnis überwacht, verschiedene Funktionen. Zum einen ist dies das Anlegen einer Datenhaltung auf dem Zielsystem unter `site/meta` (in der Struktur analog zum Document-Root des Zielsystems, wird genutzt von der SiteActive-Funktion). Zum anderen ist das das Triggern der SiteActive-Funktion, in dem über die Systemdienste geprüft wird, ob das soeben freigeschaltete Dokument in einem Alarmverzeichnis liegt. Um also ein Imperia-Zielsystem mit den Standard-Imperia-Funktionen einzurichten, muss sowohl der HTML- als auch der Notification File-Trigger für dieses Zielsystem eingerichtet werden.

Weitere Auswahl-Möglichkeiten (RDBMS-Export, Tamino-Export, XML-Export) stehen nach der Installation zusätzlicher Plug-Ins zur Verfügung. Bitte beachten Sie hierzu die gesonderte Dokumentation (Kontakt: support@imperia.net).

Über die Drop-Down-Box im Abschnitt **Übertragungsprotokoll** wird eingestellt, auf welche Art und Weise die Dateien auf das entsprechende Zielsystem übertragen werden. Abhängig von der Auswahl in dieser Drop-Down-Box werden unterschiedliche Eingabefelder in diesem Abschnitt angezeigt.

Im Abschnitt **Zieldatei** wird der Ort auf dem Zielsystem bestimmt, an dem Dokumente abgelegt werden. Als Standard wird dieser Ort aus dem Document-Root und dem Wert der Meta-Variablen `directory` und `filename` der Dokumente gebildet. In der Eingabezeile können alle Meta-Variablen der Dokumente mit XX-Syntax referenziert werden, um den Ablageort auf dem Zielsystem zu bestimmen.

Im Abschnitt **Trigger** wird festgelegt, wodurch der Freischaltvorgang auf dieses Zielsystem ausgelöst werden soll. Abhängig von der Auswahl der Drop-Down-Box werden unterschiedliche Eingabefelder in diesem Abschnitt des Dialogs angezeigt. Folgende Möglichkeiten stehen zur Verfügung:

Immer

Der Freischalt-Trigger wird immer ausgeführt, wenn ein Dokument freigeschaltet wird.

Metafeld

Der Freischalt-Trigger wird ausgeführt, wenn ein bestimmtes Metafeld einen definierten Wert hat. Das betreffende Metafeld und der entsprechende Wert werden in die Eingabefelder **Feldname** und **Wert** eingegeben. Die Werte müssen als reguläre Ausdrücke eingegeben werden und werden mit Hilfe von Pattern-Matching überprüft. Ein genauer Match wird beispielsweise durch Verankerung erreicht: `^dies muss genau matchen$`.

Benutzerprofil

Der Freischalt-Trigger wird ausgeführt, wenn das Profil des freischaltenden Benutzers ein Feld mit einem bestimmten Wert enthält. Das betreffende Feld des Benutzerprofils und der entsprechende Wert werden in die Eingabefelder **Feldname** und **Wert** eingetragen. Die Werte müssen als reguläre Ausdrücke eingegeben werden und werden mit Hilfe von Pattern-Matching überprüft. Ein genauer Match wird durch Verankerung erreicht: `^dies muss genau matchen$`. Ein klassische Beispiel hierbei ist die Verwendung des Feldnamens `username`. Andere Möglichkeiten bieten die Felder `Email-Adresse` oder `User-ID`.

Die Eingaben werden über den Button **Okay** am unteren Rand gespeichert. Der Button **Schließen** schließt den Dialog, ohne Eingaben zu speichern.

3.7.2 Freischalt-Trigger bearbeiten

Um einen bestehenden Freischalt-Trigger zu ändern, wird in der Liste der eingerichteten Freischalt-Trigger der Button **Bearbeiten** in der entsprechenden Zeile ausgewählt.

1. Öffnen Sie den Konfigurationsdialog für Freischalt-Trigger über den Menüpunkt **Zusätzliche Zielsysteme** im Ordner **System**.

Es wird eine Liste mit allen eingerichteten Freischalt-Triggern angezeigt.

2. Klicken Sie in der entsprechenden Zeile der Liste auf den Button **Bearbeiten**.

Es wird ein weiteres Fenster geöffnet, in dem die Einstellungen des Triggers geändert werden können.

3. Ändern Sie die Einträge in den einzelnen Abschnitten Ihren Wünschen entsprechend.
4. Der Button **Okay** speichert die Änderungen. Der Button **Schließen** schließt das Fenster, ohne Änderungen zu speichern.

Die Änderung eines Freischalt-Triggers ist damit abgeschlossen.

3.7.3 Freischalt-Trigger löschen

Um einen bestehenden Freischalt-Trigger zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie den Konfigurationsdialog für Freischalt-Trigger über den Menüpunkt **Zusätzliche Zielsysteme** im Ordner **System**.

Es wird ein weiteres Fenster geöffnet mit der Liste der bereits eingerichteten Freischalt-Trigger.

2. Klicken Sie neben dem Freischalt-Trigger, den Sie löschen möchten, auf den Button **Löschen**.

Der ausgewählte Freischalt-Trigger wird gelöscht und die Liste wird aktualisiert.

3.7.4 Die Darstellung beim Freischalten ändern

Insbesondere bei der Verwendung zahlreicher Freischalt-Trigger kann die Darstellung der Abfrage der einzelnen Trigger sehr umfangreich werden. Mit der System.conf-Variablen **DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER** kann die Darstellung angepasst werden. Informationen hierzu finden Sie im Anhang in Abschnitt 13.4.2.5 **Variable DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER** auf Seite 209.

3.8 Dokumentenhaltung

Imperia 8 unterstützt neben der Dokumentenhaltung im Dateisystem auch eine Reihe von SQL-Datenbanken (RDBMS):

kommerzielle Produkte: Oracle 8i (und höher)

MS-SQL ab Version 2005

frei verfügbare Produkte: PostgreSQL 8.X (und höher)

MySQL ab Version 5.X

Tabelle 3.1. SQL-Datenbanken für die Dokumentenhaltung

Weitere Datenbanken wie Informix oder SyBase können auf Anfrage und nach entsprechenden kostenpflichtigen Anpassungen zur Dokumentenhaltung genutzt werden.

Eine Beschreibung der Tabellen und deren Attribute finden Sie im Anhang in Abschnitt 13.2 **Dokumentenhaltung in einer Datenbank** auf Seite 198.

Für die nachfolgenden Kapitel setzen wir voraus, dass sowohl die Installation des RDBMS-Servers als auch des Clients bereits erfolgreich abgeschlossen wurde.

3.8.1 Unterschiede zum Dateisystem und Einschränkungen

Dokumente auf Zielsystem bearbeiten

Bei einer Dokumentenhaltung in einem RDBMS ist im Gegensatz zur Dokumentenhaltung im Dateisystem eine Einschränkung der Suche über URI-Maske mit regulären Ausdrücken nur mit (Teil-)Strings der URI möglich. Man kann also eine Zeichenfolge eingeben, die der URI enthält. Hierbei sind auch Slashes erlaubt.

Schreibtisch

Bei Dokumentenhaltung in einem RDBMS ist die Sortierung der angezeigten Dokumente nicht nach allen Kriterien möglich. Hierzu gehört zum Beispiel UID. Für einige Produkte ist die Realisierung der Sortierung nach allen Kriterien geplant (z.B.: PostgreSQL), für andere nicht (z.B.: MySQL), da dies große Performance-Einbußen mit sich bringen würde.

3.8.2 Konfiguration in Imperia

Nach der Installation ist die Dokumentenhaltung in Imperia auf das File65-Plug-In eingestellt. Im Menüpunkt **Dokumentenhaltung** der Systemadministration kann ein anderes Plug-In ausgewählt werden. Die verschiedenen Arten der Dokumentenhaltung werden durch unterschiedliche Plug-Ins realisiert.

Folgende Plug-Ins stehen zur Verfügung:

- File65
- MySQL
- PostgreSQL
- ODBC_MSSQL
- Oracle

Der Konfigurationsdialog der Dokumentenhaltung in Imperia wird über den Menüpunkt **Dokumentenhaltung** im Ordner **System** aufgerufen.

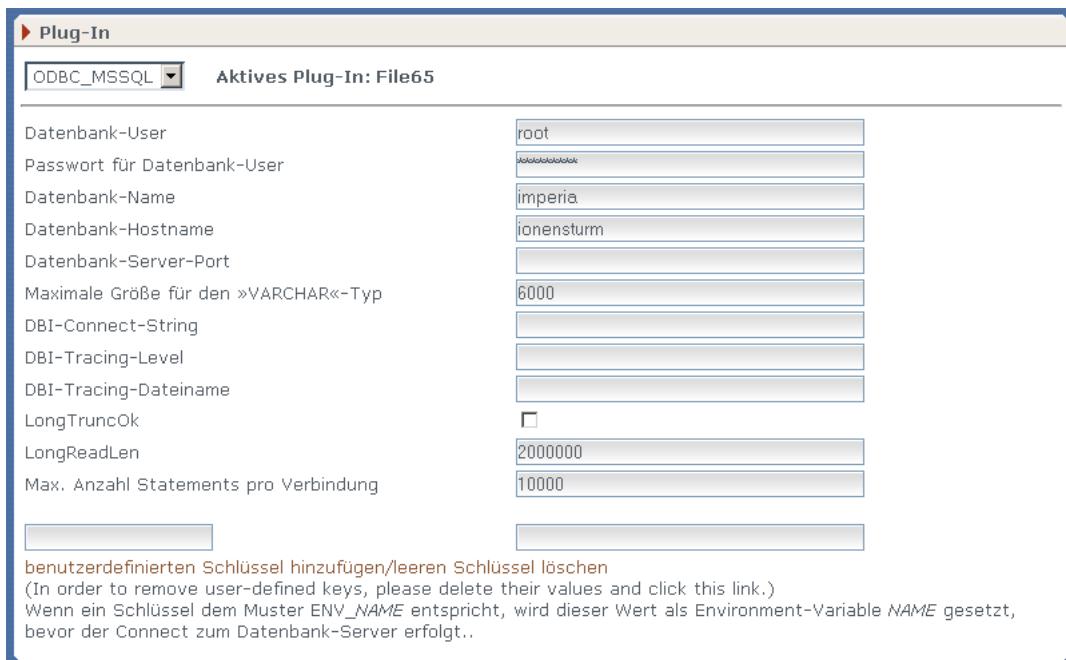


Abb. 3.9: Dialog Dokumentenhaltung



Hinweis:

Die in diesen Dialog eingegebenen Parameter werden in der Datei /site/config/repositories/default.conf gespeichert.

Der Dialog ist in zwei Abschnitte gegliedert:

Auswahl

In diesem Abschnitt wird das zu verwendende Plug-In über die Drop-Down-Box ausgewählt. Neben der Drop-Down-Box wird das momentan aktive Plug-In angezeigt.

Konfigurationsabschnitt

Dieser Abschnitt muss für jedes Plug-In gesondert bearbeitet werden. Er enthält die Eingabefelder für die Parameter, die für ein Plug-In definiert werden. In der Abbildung werden die Eingabefelder für das PostgreSQL-Plug-In angezeigt.

Ferner können im unteren Teil des Abschnittes benutzerdefinierte Schlüssel hinzugefügt werden. In die Eingabefelder müssen folgende Parameter eingetragen werden:

Datenbank-Name (Pflichteintrag!)

Name der zu verwendenden Datenbank. Diese Datenbank muss manuell angelegt werden.

Datenbank-User (Pflichteintrag!)

Name des Users, der die Datenbank benutzen darf. Dieser User muss manuell angelegt werden und die passenden Rechte für die unter „Datenbank-Name“ angegebene Datenbank besitzen.

Passwort für den Datenbank-User (Pflichteintrag!)

Passwort des Datenbank-Users.

Datenbank-Hostname (Pflichteintrag, wenn kein DBI-Connect-String definiert ist!)

Name des Hosts, auf dem die Datenbank installiert ist. Gegebenenfalls muss die dazugehörige Domäne angegeben werden.

Datenbank-Server-Port

Optional. Diesen Port nur angeben, wenn er sich vom jeweiligen Standard-Port des RDBMS unterscheidet.

Maximal-Größe für den Typ "TEXT"

Optional. Maximal mögliche Eingabe in eine Spalte vom ANSI-SQL-Typ TEXT bzw. für MSSQL-Server VARCHAR. Dieser Wert kann bei von Imperia unterstützten RDBMS leer bleiben. Je näher dieser Wert aber dem tatsächlich maximal möglichen kommt, desto besser ist die Performance, da dann bestimmte Daten weniger fragmentiert werden.

DBI-Connect-String

Mit diesem Parameter können die oben einzeln genannten Parameter auch als String übergeben werden. Dieser Parameter hat Vorrang vor allen anderen und ist auch alleine gültig. Wird hier etwas eingetragen, werden alle oberhalb dieses Eingabefeldes eingetragenen Werte nicht berücksichtigt.

DBI-Tracing-Level

Dies ist der Grad des Informationsgehalts des DBI-Traces. Je höher die hier eingetragene Ziffer ist, desto mehr Informationen werden in der Trace-Log-Datei gespeichert. Die Standardeinstellung sollte 0 (kein Trace) sein. Für die Entwicklung neuer Datenbank-Plug-Ins empfehlen wir die Einstellung 3, da die Einstellungen 1 und 2 aufgrund eines internen Problems des DBI-Moduls nicht korrekt funktionieren.

DBI-Tracing-Dateiname

Dies sind der absolute Pfad und Dateiname der Trace-Log-Datei.

LongTruncOk

Sollte nicht gesetzt werden, da im Fehlerfall lange Strings beschnitten werden, ohne dass eine Fehlermeldung auftritt.

LongReadLen

Dies ist die maximale Länge eines Strings, der gelesen werden kann. Defaultwert ist 2 Mio.

Max. Anzahl Statements pro Verbindung

Dieser Punkt gibt die maximale Anzahl der SQL-Statements an, bevor die Verbindung zur Datenbank wieder geschlossen und neu aufgebaut wird.

Zusätzliche Treiber-Optionen (nur PostgreSQL)

Hier können weitere Treiber-Einstellungen für das PostgreSQL-Plug-In getätigten werden.

Debugging-TTY

Dies ist eine zusätzliche Einstellung, auf welchem Screen die Debugging-Informationen ausgegeben werden sollen. Gilt nur für PostgreSQL unter UNIX®.

SID (Pflichteintrag für Oracle)

Dies ist der Name der Oracle-Instanz.

Sonstige Eingabefelder

Mit Hilfe der beiden Eingabefelder können eigene Parameter definiert und gesetzt werden.

Die eingegebene Konfiguration wird gespeichert, sobald der Button **Okay** am unteren Rand der Imperia-Oberfläche betätigt wird. Der Dialog wird dann im oberen rechten Bereich um die folgenden Links ergänzt:

Tabellen erzeugen

Mit diesem Link werden alle erforderlichen Tabellen automatisch erzeugt. Er öffnet einen weiteren Dialog, in dem der Username und das Passwort des RDBMS-Administrators bzw. eines User mit den entsprechenden Rechten eingegeben werden müssen. Lesen Sie hierzu Abschnitt 3.8.4 **Tabellen erzeugen** auf Seite 34.

SQL-Erzeugungs-Statements anzeigen (Trockenlauf)

Dieser Link dient zum Testen der Konfiguration. Imperia erzeugt die Tabellen nicht, sondern simuliert eine Erzeugung mit der eingestellten Konfiguration und testet damit die Verbindung zum RDBMS.

Tabellen löschen

Mit diesem Link werden die Tabellen gelöscht. Es wird ein Dialog geöffnet, in dem der Username und das Passwort des RDBMS-Administrators bzw. eines Users mit den entsprechenden Rechten eingegeben werden müssen. Lesen Sie hierzu Abschnitt 3.8.5 **Tabellen löschen** auf Seite 35.

SQL-Lösch-Statements anzeigen (Trockenlauf)

Dieser Link dient zum Testen der Konfiguration. Imperia löscht keine Tabellen, sondern simuliert das Löschen mit der eingestellten Konfiguration und testet damit die Verbindung zum RDBMS.

3.8.2.1 File65

Das Plug-In **File** wird in Imperia 8 nicht mehr unterstützt. Als Datenhaltungs-Plug-In, das alle Dokumente im Dateisystem auf der Festplatte des Produktionssystems ablegt, steht **File65** zur Verfügung. Es ist optimiert für eine große Anzahl von Dokumenten und bietet einen Geschwindigkeitsvorteil gegenüber dem File-Plug-In. Wenn Sie in einer älteren Imperia Version Ihre Daten mit dem File-Plug-In gespeichert haben und diese in Imperia 8 übernehmen wollen, müssen Sie zuvor eine Migration durchführen.

3.8.2.2 PostgreSQL

Das PostgreSQL-Plug-In bietet eine Schnittstelle zu einer PostgreSQL-Datenbank. Voraussetzung für Imperia ist die PostgreSQL-Version 8.x.

3.8.2.3 ODBC_MSSQL

Das ODBC_MSSQL-Plug-In bietet eine Schnittstelle zu einer MSSQL-Datenbank über ODBC.

3.8.2.4 Oracle

Das Oracle-Plug-In bietet eine Schnittstelle zu einer Oracle-Datenbank.

3.8.3 Konfiguration der Dokumentenhaltungsschnittstelle

Voraussetzung für die Konfiguration der Dokumentenhaltung in einer SQL-Datenbank ist die bereits erfolgte Installation folgender Perl-Module, die nicht Bestandteil der Standard-Perl-Distribution sind:

3.8.3.1 Notwendige Perl-Module

DBI

Dieses Modul (Dateiname DBI.pm) kann für UNIX®-Betriebssysteme vom CPAN heruntergeladen und installiert werden. Für Windows®-Betriebssysteme wird es von ActiveState bereitgestellt und kann mit Hilfe von ppm direkt heruntergeladen und installiert werden.



Achtung:

Verwenden Sie unter keinen Umständen die DBI-Version 1.26, da sie Fehler enthält, die einen reibungslosen Betrieb verhindern.



Hinweis:

Wenn Sie eine Oracle-Datenbank verwenden wollen, muss mindestens die DBI-Version 1.23 installiert werden. Eventuell muss die Datei DBI.pm angepasst werden, um volle Kompatibilität mit Imperia 8 zu erzielen:

Fügen Sie im BEGIN-Block (z.B.: bei Version 1.28 von DBI.pm in Zeile 11) zur Versionsnummer Anführungszeichen hinzu:

```
$DBI::Version = "1.28"; #...
```

RDBMS-spezifische Treiber

Abhängig von der Datenbank, die verwendet werden soll, muss ein weiterer spezifischer Treiber installiert werden. Die Treiber erhält man auf die gleiche Weise wie das DBI-Modul (CPAN bzw. ppm).

Die Treiber haben folgende Bezeichnungen:

RDBMS	Modulname
PostgreSQL	DBD::Pg
Oracle	DBD::Oracle
MySQL	DBD::MySQL
MSSQL	DBD::ODBC

Tabelle 3.2. RDBMS spezifische Treiber



Tipp

Ein umfassendes Repository mit Perl-Modulen für ActiveState Perl finden Sie unter

<http://theoryx5.uwinnipeg.ca/ppmpackages> (Perl-Version 5.6.X)
<http://theoryx5.uwinnipeg.ca/pmps> (Perl-Version 5.8.X)

Wenn Sie die Module nicht direkt mit ppm auf dem betreffenden System installieren können oder möchten, besteht auch die Möglichkeit, diese zunächst auf einem anderen Rechner herunterzuladen.

*In den o.a. Verzeichnissen finden Sie jeweils die *.ppd-Datei für die Installation der einzelnen Module. Die eigentlichen Module liegen als Tarball unter folgenden URLs:*

<http://theoryx5.uwinnipeg.ca/ppmpackages/x86> (Perl-Version 5.6.X)
<http://theoryx5.uwinnipeg.ca/pmps/x86> (Perl-Version 5.8.X)

Im Folgenden finden Sie eine Aufstellung der Treiber, die auf dem CPAN zur Verfügung stehen (xxxx steht für die Versionsnummer):

RDBMS	DBD-Treiber
PostgreSQL	DBD-Pg-xxxx.tar.gz
Oracle	DBD-Oracle-xxxx.tar.gz
MySQL	DBD-mysql-xxxx.tar.gz
MSSQL	DBD-ODBC-xxxx.tar.gz

Tabelle 3.3. CPAN-Module

3.8.3.2 PostgreSQL und Linux

1. Laden Sie das Modul DBD-Pg-xxxx.tar.gz vom CPAN herunter.
2. Melden Sie sich mit root-Rechten am Rechner an und verschieben Sie das heruntergeladene Modul in ein temporäres Verzeichnis.
3. Entpacken Sie das Modul mit folgendem Befehl:

```
tar -zxvf DBD-Pg-xxxx.tar.gz
```

Es wird das Verzeichnis DBD-Pg-xxxx erzeugt.

4. Wechseln Sie in das gerade erzeugte Verzeichnis und generieren Sie das Makefile mit folgendem Befehl:

```
perl Makefile.pl
```

5. Folgen Sie der Standard-Installationsprozedur:

```
make
make test
make install
```

6. Die Installation ist abgeschlossen, wenn die folgenden beiden Dateien existieren:

```
/usr/lib/perl5/site_perl/<ihre Perlversion>/i586-linux/DBD/Pg.pm
/usr/lib/perl5/site_perl/<ihre Perlversion>/i586-linux/Pg.pm
```

Installation von PostgreSQL unter SuSE 7.1 (oder höher)

Wird eine SuSE-Distribution der Version 7.1 oder höher verwendet, kann die mit SuSE gelieferte PostgreSQL-Version verwendet werden.

Laden und entpacken Sie anschließend den Datenbank-Treiber wie oben beschrieben. Bevor der Treiber unter SuSE installiert werden kann, müssen zunächst das SuSE-Paket `postgres-devel` installiert und folgende Umgebungsvariablen `POSTGRS_LIB` und `POSTGRES_INCLUDE` gesetzt werden.

Beispiele:

```
thomas@work:/tmp/DBPD-Pg-1.01 > export POSTGRES_INCLUDE=/usr/include/pgsql/
thomas@work:/tmp/DBPD-Pg-1.01 > export POSTGRES_LIB=/usr/lib/
```

Danach kann die Installation wie oben beschrieben durchgeführt werden. Legen Sie anschließend einen User für die Datenbank an. Loggen Sie sich als Superuser (User `postgres`) der Datenbank ein und legen Sie einen neuen User mit dem Recht, eine Datenbank zu erzeugen, an.



Achtung:

Der Superuser der Datenbank erhält bei der Installation ein dem Autor unbekanntes, nicht standardisiertes Passwort. Ändern Sie dieses Passwort, um neue Datenbank-User anlegen zu können.

3.8.3.3 PostgreSQL und Windows

- Starten Sie eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install DBD-Pg
```

Nach erfolgreicher Installation können Sie die Konfiguration und die Erzeugung der Tabellen in Imperia vornehmen. Eventuell muss die Datei `DBI.pm` in den Perlpath aufgenommen werden.

Wenn Sie für ppm kein Repository eingerichtet haben, in dem dieses Modul verfügbar ist, können Sie es aus dem in Abschnitt 3.8.3.1, „RDBMS-spezifische Treiber“ erwähnten Repository herunterladen.

3.8.3.4 Oracle und Linux

Bevor der Oracle-Treiber installiert werden kann, muss der Oracle-Client installiert werden. Hierzu verwendet man die gelieferte Oracle-CD, startet das Installationsprogramm, wählt den Client aus und folgt den Anweisungen. Lediglich der Port, den Sie bei Ihrem Systemadministrator erfragen können, muss eventuell angepasst werden.

Nach erfolgreicher Installation des Oracle-Clients kann der Treiber installiert werden.

1. Laden Sie das Modul `DBD-Oracle-xxxx.tar.gz` vom CPAN herunter.
2. Melden Sie sich mit root-Rechten am Rechner an und verschieben Sie das heruntergeladene Modul in ein temporäres Verzeichnis.
3. Entpacken Sie das Modul mit folgendem Befehl:

```
tar -zxvf DBD-Oracle-xxxx.tar.gz
```

Es wird das Verzeichnis `DBD-Oracle-xxxx` erzeugt.

4. Wechseln Sie in das gerade erzeugte Verzeichnis und generieren Sie das Makefile mit folgendem Befehl:

```
perl Makefile.pl
```

5. Folgen Sie der Standard-Installationsprozedur:

```
make  
make test  
make install
```

3.8.3.5 MySQL und Windows®

- Starten Sie eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
ppm install DBD-mysql
```

Nach erfolgreicher Installation können Sie die Konfiguration und die Erzeugung der Tabellen in Imperia vornehmen. Eventuell muss die Datei `DBI.pm` in den Perlpath aufgenommen werden.

3.8.3.6 MySQL und Linux

Bei der Installation von MySQL müssen Sie auf die Version MySQL 5.X zurückgreifen, die Mindestvoraussetzung für Imperia 8.5 ist.

Nach erfolgreicher Installation von MySQL installieren Sie den DBD-mysql-xxxx-Treiber:

1. Melden Sie sich mit root-Rechten am Rechner an und verschieben Sie das heruntergeladene Modul in ein temporäres Verzeichnis.
2. Das Modul wird mit folgendem Befehl entpackt:

```
tar -zxvf DBD-mysql-xxxx.tar.gz
```

Es wird das Verzeichnis DBD-mysql-xxxx erzeugt.

3. Wechseln Sie in das erzeugte Verzeichnis und generieren Sie das Makefile mit folgendem Befehl:

```
perl Makefile.pl
```

4. Folgen Sie der Standard-Installationsprozedur:

```
make  
make test  
make install
```

3.8.3.7 SQL-Server und Windows®

1. Installieren Sie den ODBC-Treiber von ActiveState mit Hilfe von ppm .

```
ppm install DBD-ODBC
```

2. Richten Sie anschließend unter

Start/Einstellungen/Systemsteuerung/Verwaltung/Datenquellen ODBC

den SQL-Server als System-DSN ein.

3. Beantworten Sie die Frage "Wie soll SQL-Server die Authentizität des Benutzernamens bestätigen?" mit der zweiten Antwort "Mit SQL-Server Authentifizierung anhand des vom Benutzer eingegebenen Benutzernamens und Kennworts".

Der Datenquellen-Name, der hier vergeben wird, muss in der Konfiguration der Imperia-Dokumentenhaltung als **Database Name** angegeben werden.

3.8.4 Tabellen erzeugen

Alle notwendigen Tabellen werden automatisch erzeugt. Beim Klick auf den Link **Tabellen erzeugen** erscheint ein weiterer Dialog, in dem der Name und das Passwort des RDBMS-Administrators bzw. eines Users mit entsprechenden Rechten eingegeben werden muss. Im Feld **Aktion** muss der Eintrag **Tabellen erzeugen** ausgewählt sein.

Wurden alle Angaben gemacht, können die Tabellen mit einem Klick auf den Button **Okay** erzeugt werden. Nach erfolgreicher Durchführung wird eine Liste aller erzeugten Tabellen angezeigt.

3.8.5 Tabellen löschen

Alle Imperia-Tabellen werden automatisch gelöscht. Dabei gehen alle Daten verloren. Ein Klick auf den Link **Tabellen löschen** öffnet einen weiteren Dialog, in den der Name und das Passwort des RDBMS-Administrators bzw. eines Users mit entsprechenden Rechten eingegeben werden muss.



Hinweis:

*Bleiben die Felder **Username** und **Passwort** leer, versucht Imperia die Tabellen mit dem bereits in der Dokumentenhaltungskonfiguration angegebenen Usernamen/Passwort zu erzeugen. Hat dieser User die entsprechenden Rechte nicht, können die Tabellen nicht angelegt werden.*

Im Feld **Aktion** muss der Eintrag **Tabellen löschen** ausgewählt sein.

Wurden alle Angaben gemacht, können die Tabellen mit einem Klick auf den Button **Okay** gelöscht werden. Nach erfolgreicher Durchführung wird eine Liste aller gelöschten Tabellen angezeigt.

3.9 One-Click-Edit

Mit One-Click-Edit (OCE) können Sie ein veröffentlichtes Dokument, das Sie auf einem Zielsystem betrachten direkt von dort aus zum Bearbeiten öffnen. Im Hintergrund sorgt Imperia für die Einhaltung des zugehörigen Workflows und der Freigabemechanismen. Sie brauchen lediglich eine Browser-Erweiterung zu installieren, um OCE nutzen zu können.

Imperia erkennt, ob ein User bereits angemeldet ist und bietet bei entsprechender Konfiguration automatisch die Login-Maske an. Im Folgenden wird die Konfiguration der verfügbaren Funktionen der OCE-Toolbar beschrieben.

Innerhalb eines Imperia-Dokuments werden bei entsprechender Programmierung des Templates die NodeID und die System-ID verschlüsselt gespeichert. Hierzu wird entweder ein eigener Namespace oder ein META-Tag verwendet. Bei der META-Tag-Variante ist außerdem die Speicherung von Pfad und Dateiname der betreffenden Seite möglich. So lassen sich beispielsweise Copy-Seiten unabhängig vom Master-Dokument oder anderen Kopien vom Live-System entfernen. Lesen Sie hierzu auch den Abschnitt "**One-Click-Edit**" im Programmierhandbuch und Abschnitt 13.4.22.1 **Variable ONECLICKEDIT** auf Seite 224.

Bei installierter OCE-Toolbar, gleicht One-Click-Edit nach der Anmeldung des Benutzers am zugehörigen Entwicklungs-System bei jedem Seitenaufruf automatisch die Benutzer-Rechte sowie Dokumenten-Informationen mit dem Develop-Server ab, wenn die aufgerufene Seite von diesem System stammt. Der Benutzer kann immer nur die Funktionen der Toolbar aufrufen bzw. sehen, zu deren Ausführung er für das jeweilige Dokument berechtigt ist.



Wichtig:

Beachten Sie, dass beim Kopieren eines Systems, etwa zu Testzwecken, die SysID ebenfalls mitkopiert wird. Dies führt zu Problemen, wenn Benutzer auf dem Testsystem Content freischalten oder mit One-Click-Edit Dokumente bearbeiten. Da die kopierte SysID eingelesen wird, arbeiten die Benutzer nicht auf dem Testsystem sondern auf dem Produktivsystem, von dem die Kopie angefertigt wurde.

Damit dieser Fehler nicht auftritt, müssen Sie in der Kopie eine neue SysID erzeugen. Gehen Sie hierzu folgendermaßen vor:

1. *Löschen Sie die Datei site/secure/sysid in Ihrem Testsystem. Melden Sie sich anschließend an dem System an. Dabei wird automatisch eine neue SysID generiert.*
2. *Reparieren Sie den kompletten Content Ihres Testsystems, damit die neue SysID in die Dokumente eingebunden wird.*
3. *Nun können Sie gefahrlos auf dem Testsystem arbeiten.*

OCE wird im Dialog **One-Click-Edit-Konfiguration** konfiguriert, der über den Menüpunkt **One-Click-Edit** im Ordner **System** geöffnet wird. Der Dialog enthält für jede mögliche OCE-Funktion einen eigenen Abschnitt.

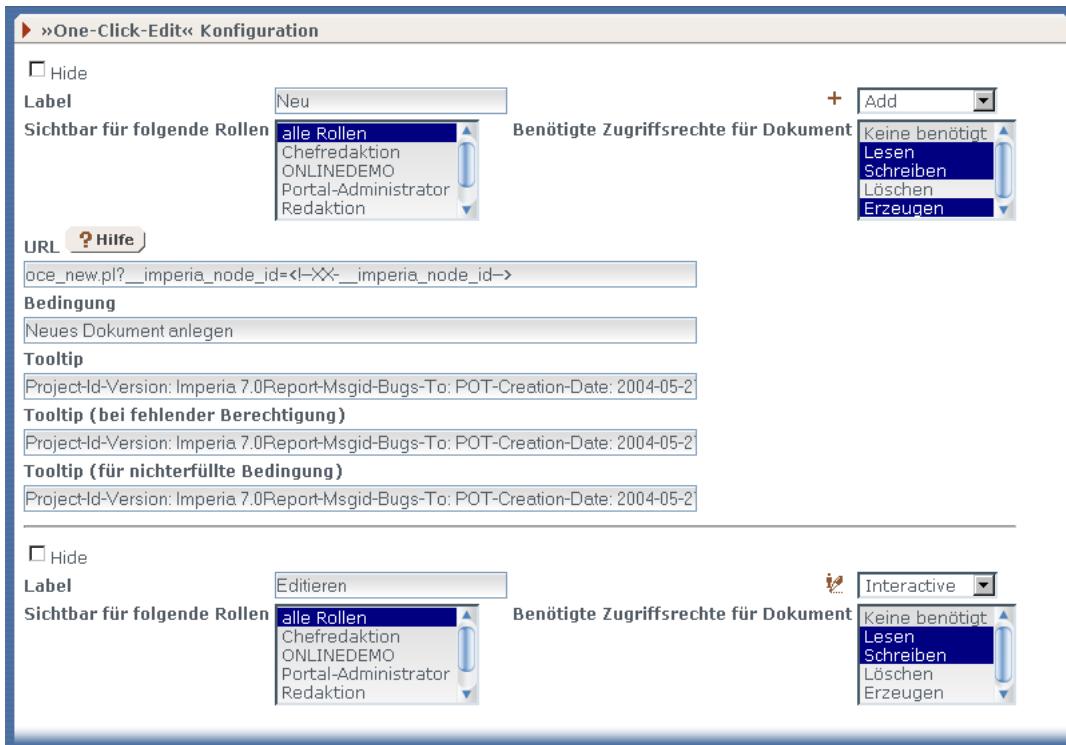


Abb. 3.10: Dialog OCE-Konfiguration

Der Dialog gliedert sich in zwei Abschnitte. Im oberen Teil wird unter anderem eingestellt, welche Rollen zur Verwendung dieser OCE-Funktion berechtigen. Ferner wird eingestellt, welche Dokumenten-Zugriffsrechte benötigt werden.

Im unteren Abschnitt werden die URL, eine Bedingung sowie verschiedene Tooltips eingegeben.

Die Elemente des Dialogs bedeuten im Einzelnen:

Hide

Mit dieser Checkbox wird die Funktion global aus- bzw. eingeschaltet. Wird die Checkbox markiert, steht die Funktion nicht zur Verfügung.

Icon

In dieser Liste wird das Icon ausgewählt, das in der OCE-Toolbar angezeigt werden soll.

Label

In dieses Eingabefeld wird der Titel der Funktion eingegeben. Der Titel erscheint in der OCE-Toolbar neben dem Icon.

Sichtbar für folgende Rollen

In diesem Feld werden die Rollen markiert, die zur Verwendung der OCE-Funktion berechtigen. Mehrere Rollen werden mit Hilfe der Strg-Taste markiert.

Benötigte Zugriffrechte für Dokument

In diesem Feld werden die Zugriffsrechte für ein Dokument bestimmt, die der User mindestens haben muss, um die Funktion ausführen zu können. In der Hilfe werden die Rechte für eine Funktion empfohlen.

URL

In diesem Feld wird die URL eingegeben, den die OCE-Toolbar an das Produktionssystem schicken soll. Er besteht aus dem Namen des Skripts sowie verschiedenen Parametern. Lesen Sie hierzu auch Abschnitt 3.9.2 **OCE-Skript hinzufügen** auf Seite 37.

Hilfe

Dieser Button zeigt einen Hilfetext zu der ausgewählten OCE-Funktion. Die Hilfe basiert auf den PODs der entsprechenden Imperia-Skripte.

Bedingung

Definieren Sie eine optionale Bedingung, die erfüllt sein muss, damit die OCE-Funktion in der OCE-Toolbar erscheint. Die Notation erfolgt in der Template-Syntax der IF-Abfragen, z.B.: "<!--XX-directory-->" NE "/no_quick_edit").

Tooltip

Hier wird der Text eingegeben, der angezeigt wird, wenn der User den Mauszeiger über dem Icon in der OCE-Toolbar hält.

Tooltip (bei fehlender Berechtigung)

Hier wird der Text eingegeben, der angezeigt wird, wenn der User den Mauszeiger über dem Icon in der OCE-Toolbar hält, er aber nicht über die notwendigen Rechte verfügt.

Tooltip (für nichterfüllte Bedingung)

Hier wird der Text eingegeben, der angezeigt wird, wenn der User den Mauszeiger über dem Icon in der OCE-Toolbar hält, jedoch die eingestellte Bedingung nicht erfüllt ist.

3.9.1 One-Click-Edit und Copy-Seiten

Normalerweise arbeitet One-Click-Edit auf Dokumentenbasis. Das bedeutet für Dokumente mit Copy-Pages, dass eine OCE-Funktion, die Sie auf einer Copy-Seite ausführen, sich immer auf das gesamte Dokument auswirkt. Beispielsweise löschen Sie also im Normalfall mit der Funktion **Live-Löschen** nicht nur die jeweilige Kopie vom Live-System, sondern auch alle anderen existierenden Kopien des Dokuments. Durch eine besondere Einbindung der OCE-Informationen ins Template eines Dokuments oder durch eine Variable zur Systemkonfiguration können Sie dieses Verhalten auf eine dateibasierte Verarbeitung umstellen. Für weitere Einzelheiten hierzu lesen Sie bitte Abschnitt 13.4.22.1 **Variable ONECLICKEDIT** auf Seite 224 und den Abschnitt **One-Click-Edit** in der Syntaxreferenz im Kapitel **Templates** des Programmierhandbuchs.

3.9.2 OCE-Skript hinzufügen

Neben den mitgelieferten OCE-Skripten können auch andere Imperia-Skripte oder eigene Skripte als OCE-Skript verwendet werden. OCE-Skripte werden im CGI-BIN-Verzeichnis erwartet. Als Namenskonvention gilt, dass die Imperia-OCE-Skripte mit `oce_` beginnen müssen.

Am unteren Ende des Dialogs **One-Click-Edit-Konfiguration** befindet sich ein leerer Abschnitt, in den ein weiteres OCE-Skript eingetragen werden kann. Sobald eine neue Skript-Konfiguration gespeichert wurde, erscheint automatisch ein neuer leerer Abschnitt. Am Beginn dieses Kapitels werden die einzelnen Felder des Dialogs und die notwendigen Eingaben beschrieben.

Wir möchten hier auf das Eingabefeld **URL** eingehen, in dem die URL festgelegt wird, den der entsprechende Button der OCE-Toolbar übergibt. Es können auch Links eingegeben werden, die nicht ein Imperia-Produktionssystem ansprechen. Zum Beispiel zu einem Übersetzungsdiagramm oder einer Suchmaschine.

Eine URL, die das OCE-Skript übergibt, besteht aus dem Namen des Skripts und verschiedenen Parametern. Ein wichtiger Parameter ist die Imperia-NodeID, über die das Dokument vom Produktionssystem identifiziert wird.

3.9.3 Imperia OCE-Skripte

Für alle Imperia-OCE-Skripte ist über den Button Hilfe im Konfigurationsdialog eine englische Hilfe verfügbar. Folgende Skripte sind im Lieferumfang enthalten:

3.9.3.1 oce_edit

Dieses Skript öffnet das Dokument zum Editieren. Befindet sich das Dokument nicht im Workflow, wird die letzte gespeicherte Version aus dem Archiv importiert, bevor es zum Editieren geöffnet wird. Als Dokumenten-Zugriffsrechte sollte **Lesen** und **Schreiben** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

`__imperia_node_id`

Die NodeID des Dokuments.

`finished`

Wird dieser Parameter als `true` (oder `1`) übergeben, wird die letzte **beendete** Version und nicht die zuletzt gespeicherte Version aus dem Archiv importiert.

`__imperia_quick_edit`

Wird dieser Parameter gesetzt, werden alle Workflow-Plug-Ins, die QuickEdit nicht unterstützen, ausgelassen.

3.9.3.2 oce_clone

Dieses Skript erzeugt einen fast identischen Klon eines Imperia-Dokuments. Das neue Dokument unterscheidet sich zumindest in den Meta-Variablen `__imperia_node_id`, `__imperia_created` und `__imperia_modified`. Auch der Wert von anderen Meta-Variablen kann mit Hilfe passender Parameter verändert werden. Als Dokumenten-Zugriffsrechte sollte **Lesen**, **Schreiben** und **Erzeugen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

`__imperia_node_id`

Die NodeID des Dokuments.

`__imperia_created`, `__imperia_modified`

Diese beiden Variablen enthalten die aktuelle Zeit in Sekunden seit der Epoche (Sekunden seit 00:00:00, 01.01.1904 GMT für MacOS und 00:00:00, 01.01.1970 für UNIX®- und Windows®-Betriebssysteme).

`__imperia_uid`

Enthält die User-ID des aktuell angemeldeten Users.

`__imperia_meta_expand`

Sofern die Meta-Variable `directory` nicht vorhanden ist, enthält dieser Parameter den Wert 1, unabhängig von seinem vorhergehenden Wert. Dieser Parameter wird vom MetaEdit-Plug-In verarbeitet. Ist er vorhanden und nicht 0, wird `<!--XX-METAMODE-->` zu true expandiert. Der Parameter wird nach der ersten Verarbeitung gelöscht. Infolgedessen wird ein neues Dokument, wenn der Parameter einen wahren Wert hat, vom MetaEdit-Plug-In wie ein gerade erzeugtes Dokument behandelt. Dies wird meist für die Expansion von `<!-- XX-count-->` in Metadateien verwendet.



Hinweis:

Unter normalen Bedingungen sollte das Metafeld `directory` immer gelöscht werden, bevor ein Dokument geklont wird. Denn dann wird es durch die neu zugeordnete Rubrik initialisiert.

`__imperia_parent`

Dieser Parameter enthält die NodeID des Eltern-Dokuments.

`__imperia_children`

Dieser Parameter enthält die NodeID des Kind-Dokuments, die an die Liste aller Kind-Dokumente eines Eltern-Dokuments(!) angehängt wird.

3.9.3.3 oce_erase

Mit diesem Skript wird ein Imperia-Dokument komplett gelöscht. Es wird von allen Zielsystemen und aus dem Produktionssystem gelöscht. Ferner werden alle Archivinformationen und alle zugehörigen Objekte (Bilder etc.) gelöscht. Als Dokumenten-Zugriffsrechte sollte **Lesen**, **Schreiben**, **Erzeugen** und **Löschen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

`__imperia_node_id`

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

`force`

Wird dieser Parameter auf einen wahren Wert gesetzt, erfolgt das Löschen ohne Bestätigungsdialog. Andernfalls erscheint eine Liste aller gelöschten Objekte.

3.9.3.4 oce_delete

Mit oce_delete können Sie ein Dokument von allen Zielsystemen löschen. Das Skript kann folgende Parameter entgegennehmen:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

directory

Dies ist das Verzeichnis auf dem Zielsystem.

filename

Dies ist der Dateiname des Dokuments im Verzeichnis des Zielsystems.

copies

Wenn Sie diesen Parameter auf einen wahren Wert setzen, werden auch alle Kopien des Dokumentes entfernt.

del_develop

Wenn Sie diesen Parameter auf einen wahren Wert setzen, wird die Löschaktion auch auf dem Produktions-
system ausgeführt.

3.9.3.5 oce_history

Dieses Skript zeigt die Dokumenten-History an. Als Dokumenten-Zugriffsrechte sollte **Lesen** eingestellt
werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

3.9.3.6 oce_info

Dieses Skript zeigt eine Reihe von Informationen zum Dokument an. Als Dokumenten-Zugriffsrechte sollte
Schreiben eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

3.9.3.7 oce_linkcheck

Dieses Skript überprüft alle internen und externen Links in einem Dokument. Als Dokumenten-Zugriffsrechte
sollte **Lesen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

finished

Wird dieser Parameter auf 1 (true) gesetzt, wird die letzte beendete Version überprüft und nicht, wie stan-
dardmäßig, die letzte bearbeitete Version.

__imperia_quick_edit

Wird dieser Parameter gesetzt, werden alle Workflow-Plug-Ins, die QuickEdit nicht unterstützen, ausgelassen.

3.9.3.8 oce_reparse

Dieses Skript parst das Dokument neu, so dass Änderungen am Template übernommen werden. Optional
kann das Freischalten auf alle Zielsysteme erzwungen werden. Als Dokumenten-Zugriffsrechte sollte **Lesen**,
Schreiben und **Erzeugen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

finished

Wird dieser Parameter auf 1 (true) gesetzt, wird die letzte beendete Version als Basis für das Reparsen herangezogen und nicht, wie standardmäßig, die letzte bearbeitete Version.

publish

Wird dieser Parameter auf einen wahren Wert gesetzt, werden erfolgreich reparste Dokumente automatisch auf die Zielsysteme freigeschaltet. Dies geschieht jedoch nur, wenn das Dokument vorher schon einmal freigeschaltet wurde.

3.9.3.9 oce_unpublish

Dieses Skript entfernt das Dokument von allen Zielsystemen. Als Dokumenten-Zugriffsrechte sollte **Löschen** und **Erzeugen** eingestellt werden. Folgende Parameter können dem Skript übergeben werden:

__imperia_node_id

Dies ist die NodeID des Dokuments (z.B. /1/1/1967).

directory

Dies ist das Verzeichnis auf dem Zielsystem.

filename

Dies ist der Dateiname des Dokuments im Verzeichnis des Zielsystems.

3.10 Allgemeine Einstellungen

Über den Dialog **Allgemeine Einstellungen** werden verschiedene systemweite Einstellungen vorgenommen.

3.10.1 Konfiguration

Dieser Abschnitt dient der Navigation durch den Dialog. Mit Hilfe der Links wird auf die einzelnen Abschnitte zugegriffen.

3.10.2 Cookie-Lebensdauer

Tragen Sie hier die Gültigkeitsdauer des Imperia-Cookies in Sekunden ein. Der voreingestellte Wert ist 7200 Sekunden. Führt ein Anwender für einen Zeitraum, der länger als diese Einstellung ist, keine Aktion aus, muss er sich erneut am System anmelden.

3.10.3 Dokumenten-History

In diesem Abschnitt legen Sie die Einstellungen für die Anzeige der Dokumenten-History fest. Folgende Einstellungsmöglichkeiten stehen Ihnen zur Verfügung:

Anzeigelänge der Dokumenten-History

Die Anzahl der angezeigten Einträge. Setzen Sie den Wert auf Null, um das Feature zu deaktivieren. Dadurch wird grundsätzlich die vollständige History angezeigt.

Sichtbarkeit für Datum und Uhrzeit in der Dokumenten-History

Legen Sie fest, wer Datum und Uhrzeit in der Dokumenten-History sehen kann. Zur Auswahl stehen:

- everybody - Alle Rollen.
- nobody - Die Anzeige von Datum und Uhrzeit ist generell unterdrückt.
- superuser - Nur Benutzer mit Zugriff auf die Superuser-Rolle sehen Datum und Uhrzeit.

3.10.4 Java-VM

In diesem Abschnitt wird die Java VirtualMachine konfiguriert. Folgende Einstellungen können vorgenommen werden:

Java Executable

Geben Sie in dieses Feld den Pfad zum Java-Interpreter oder zum Java-Wrapper-Skript ein (z.B. /usr/lib/java/bin/java).

Classpath

Tragen Sie in dieses Feld die Classpath-Einstellungen für die Java VirtualMachine auf dem Imperia-Server ein. Lassen Sie dieses Feld im Zweifelsfall leer.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

3.10.5 Link-Checker

Dieser Abschnitt konfiguriert den Link-Checker für externe Links. Über die Checkbox **Link-Check mit HTTP-Requests aktivieren** wird die Funktion aktiviert. Falls in Ihrem Netzwerk ein Proxy-Server verwendet wird, können entsprechende Einstellungen in den einzelnen Eingabefeldern getätigt werden.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

3.10.6 Superuser Authentifizierungs-Plug-In

Markieren Sie die Checkbox, um das System in den Wartungsmodus mit Superuser-Zugriff zu versetzen. Lesen Sie hierzu auch Abschnitt 3.2.2 **Wartungsmodus mit Superuser-Zugriff** auf Seite 18

3.10.7 System-Sprache

In diesem Abschnitt wird die System-Sprache und das System-Charset eingestellt.

Die Standardeinstellung hierfür ist Englisch. Mit Hilfe des Pull-Down-Menüs können andere Einstellungen ausgewählt werden.

Unabhängig von der ausgewählten Sprache werden sämtliche Daten im Zeichensatz UTF-8 gespeichert und ausgeliefert. Das gilt sowohl für alle Bildschirmausgaben als auch für alle Daten, die von Benutzern eingegeben wurden.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

3.10.8 Template-Prozessor

Standardmäßig werden Skripte, Layer/ILayer, Frames und Formulare im PREVIEW und im EDIT-Modus nicht unterstützt. In diesem Abschnitt kann dieses Verhalten geändert werden. Markieren Sie die entsprechenden Checkboxen.

Speichern Sie die Einstellungen mit Hilfe des Buttons **Speichern** am unteren Rand des Dialogs.

3.10.9 Unicode in Imperia

Die Unicode-Unterstützung in Imperia hat zwei verschiedene Aspekte. Man muss unterscheiden zwischen HTML-Seiten, die die Imperia-Oberfläche bilden, und Dokumenten, die auf ein Zielsystem freigeschaltet werden.

Typische Beispiele für Oberflächen-Seiten sind das Hauptmenü, die Rubrikenansicht und der Schreibtisch. Zu den Dokumenten gehören neben den freigeschalteten Seiten auf dem Zielsystem selbst auch alle Komponenten, die bei deren Erstellung zum Einsatz kommen. Das beinhaltet Metadateien, Templates, Includes, importierte Daten aus externen Quellen und natürlich die von den Benutzern eingegebenen Daten.

Imperia speichert sämtliche Daten intern im Unicode-Zeichensatz UTF-8. Da moderne Browser und Editoren mittlerweile alle UTF-8 unterstützen, ist so gewährleistet, dass beliebige sprachspezifische Sonderzeichen korrekt ausgeliefert und dargestellt werden können.

Die in älteren Imperia-Versionen übliche Konfiguration des verwendeten Zeichensatzes kann deswegen entfallen. Der Zeichensatz UTF-8 ist auch dann die optimale Wahl, wenn Sie gegenwärtig keine Mehrsprachigkeit in Ihrem Projekt planen.



Hinweis:

Die aus älteren Imperia-Versionen bekannte Konfigurationsvariable `DEFAULT_CHARSET` hat keine Wirkung mehr.

Wenn Sie mit Imperia 8 ein neues Projekt beginnen, ist die Verwendung des UTF-8 Zeichensatzes empfehlenswert. Sollten Sie Gründe haben, einen anderen Zeichensatz einzusetzen, ist eine Konvertierung des Contents vor dem Freischalten der Dokumente notwendig.

Für diese Konvertierung stehen Ihnen in Imperia zwei Post-Convert Plug-Ins zur Verfügung:

- **Recode** - mit diesem Plug-In konvertieren Sie den Inhalt einzelner Metafelder von UTF-8 in Ihren gewünschten alternativen Zeichensatz.

- **SafeUnicode** - dieses Plug-In wandelt Sonderzeichen automatisch in HTML-Entitäten um (z.B. ü für den Buchstaben "ü").

Sollten Sie ein bestehendes Projekt migrieren und bisher einen anderen Zeichensatz als UTF-8 eingesetzt haben, nutzen Sie bitte die Migrationsskripte, die mitgeliefert werden, um die Inhalte zu konvertieren.

3.11 Konfiguration des Systems über die Benutzeroberfläche

Imperia 8 bietet Ihnen eine Schnittstelle zur Änderung sämtlicher Konfigurationsdateien. So brauchen Sie keinen Zugriff auf Dateisystemebene mehr, um Einstellungen des Systems zu ändern, die nicht direkt über die Benutzeroberfläche erreichbar sind.



Wichtig:

Lesen Sie den Anhang Abschnitt 13.4 Liste aller Variablen zur System-Konfiguration auf Seite 209 für Informationen über die Konfigurationsvariablen von Imperia.

Diese Funktion ist nur für Benutzer mit Superuser-Berechtigung zugänglich. Die Schnittstelle rufen Sie über den Menüpunkt *Hauptmenü -> System -> Konfiguration* auf.

Sie gelangen zu einer Übersicht der editierbaren Konfigurationsdateien:

Konfigurationsdateien	
config/config.conf	
config/editmode.conf	
config/fts.conf	
config/index.conf	
config/java.conf	
config/javavm.conf	
config/LDAP.conf	
config/logrotate.conf	
config/mdb_db.conf	
config/mdb_extra.conf	
config/mdb_ext_server.conf	
config/mdb_search.conf	
config/quick-edit.conf	
config/search.conf	
config/smart-meta.conf	
config/system.conf	
config/system_files.conf	

Abb. 3.11: Übersicht der editierbaren Konfigurationsdateien

Klicken Sie auf einen der Dateinamen, um sich den Inhalt der betreffenden Datei anzeigen zu lassen.

► Anzeige von /config/mdb_search.conf	
Schlüssel	Wert(e)
Über diese Einträge ist eine Suche möglich Format: FELDNAME=Angezeigter Text	
NAME	Name
COPYRIGHT	Copyright
COMMENT1	Kommentar 1
COMMENT2	Kommentar 2
COMMENT3	Kommentar 3
OBJURL	URL
USED	Used Counter
SELLPRICE	Verkaufspreis
VIEWPRICE	View.Preis
DATATYPE	Dateiendung
SERVER	Server
KEYWORDS	Keywords

Bearbeiten

Abb. 3.12: Anzeige des Inhalts einer Konfigurationsdatei

In dieser Ansicht ist der Inhalt der Datei nicht editierbar. Über den Link **Bearbeiten** können Sie die Datei jedoch zur Bearbeitung öffnen.

Zur Bearbeitungsansicht einer Konfigurationsdatei gelangen Sie auch, wenn Sie in der Übersicht auf das Icon neben dem Dateinamen klicken. Ein Beispiel für die Bearbeitungsansicht zeigt die nächste Abbildung:

► Datei: »system.conf«

```

# -----
# IMPERIA 7 CONFIGURATION FILE
# -----


"OPERATING-SYSTEM" = "UNIX"
"DISABLE-FLOCK" = "0"
"DOCUMENT-ROOT" = "/usr/local/share/imperia/dev_ionensturm"
"CGI-DIR" = "/usr/local/share/imperia/dev_ionensturm/cgi-bin"
"CGI-BIN" = "/cgi-bin"
"DEMOMODE" = "no"
"DISABLE-FORTUNE-COOKIE" = "1"

"LOCK_SH" = "1"
"LOCK_EX" = "2"
"LOCK_NB" = "4"
"LOCK_UN" = "8"

"DEFAULT_LOCALE" = "de"

"DEFAULT_CHARSET" = "UTF-8"
"live-url" = "http://ionensturm.imperia.net"
"COOKIE_TTL" = "720000"
"transfer" = "Filecopy"
"WORDMODUL_USER" = "Imperia AG"

"DOCTREE_PLUGIN" = "File65"
"MD-WINDOW-HEIGHT" = "800"
"MD-WINDOW-HEIGHT" = "1000"

```

Speichern Schließen

Abb. 3.13: Der Editor für Konfigurationsdateien

Der Editor gleicht in seiner Funktion den Editoren für Flexmodule, Codeincludes und Dynamische Module.

3.11.1 Konfiguration der Schnittstelle

Per Default sind sämtliche Konfigurationen editierbar. Sie können diese Auswahl einschränken, indem Sie die Datei config.conf editieren.

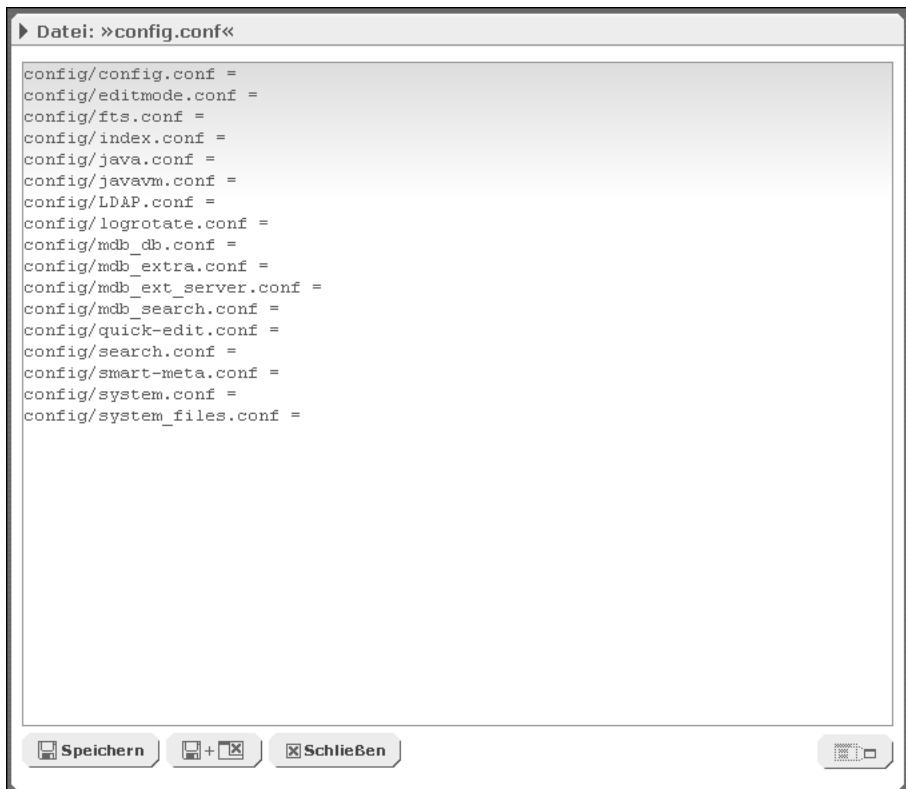


Abb. 3.14: Die Datei config.conf

Diese Datei enthält eine Liste der Konfigurationsdateien, die über die Benutzeroberfläche bearbeitet werden können. Um eine Datei aus dieser Liste zu entfernen, löschen Sie den Dateinamen aus der Liste oder setzen Sie das Kommentarzeichen '#' davor.

Umgekehrt können Sie eine Datei editierbar machen, indem Sie sie der Liste hinzufügen oder ein ggf. vorhandenes Kommentarzeichen vor dem Dateinamen entfernen.

3.12 Konfiguration der Link-Konsistenzprüfung

Die Konfiguration der Link-Konsistenzprüfung nehmen Sie in der Datei linkmanager.conf im Verzeichnis site/config Ihrer Imperia-Installation vor. Sie können die Links auf mehreren Systemen prüfen lassen. Die Einstellungen für ein System notieren Sie immer innerhalb eines Blocks, der durch die Tags

```
<SERVER MeinServer>  
[Einstellungen für das System]  
</SERVER>
```

eingeschlossen wird. Anstelle des Platzhalters MeinServer tragen Sie eine innerhalb der Konfigurationsdatei eindeutige Bezeichnung für das betreffende System ein.

Diese werden immer nach dem nachfolgenden Schema notiert:

Direktive = "Wert"

Im Folgenden eine Erläuterung der einzelnen Direktiven für die Konfiguration:

SERVERNAME

Tragen Sie hier einen Namen für das System ein. Dieser Name steht in der Titelzeile des Berichtes mit den Ergebnissen der Link-Konsistenzprüfung für das betreffende System.

```
SERVERNAME = "Live-System"
```

SERVER_URL

Tragen Sie hier die URL des Systems ein, auf die die Konsistenzprüfung HTTP-Requests zur Prüfung der Links ausführen soll.

```
SERVER_URL = "http://www.myserver.de"
```

SETTYPE_HTML

Geben Sie Dateien an, die die Konsistenzprüfung dem Typ HTML zugehörig einstufen soll. Diese Angabe wird als regulärer Ausdruck auf die Dateinamen angewendet. Alternative Muster werden durch das Pipe-Zeichen ("|") voneinander abgetrennt.

```
SETTYPE_HTML = "\..?htm.?|\..?php"
```

SETTYPE_CSS

Diese Direktive enthält eine Auflistung der Dateien, die als Cascading-Stylesheet-Dateien behandelt werden sollen. Geben Sie die entsprechenden Dateiendungen als regulären Ausdruck an. Die alternativen Dateiendungen werden durch das Pipe-Zeichen ("|") voneinander abgetrennt.

```
SETTYPE_CSS = "\.css/i"
```

EXCLUDE_FILE

Tragen Sie hier einen regulären Ausdruck für die Dateien ein, die von der Link-Konsistenzprüfung nicht kontrolliert werden sollen. Dateien, deren Name auf das Muster passt, werden ausgelassen.

```
EXCLUDE_FILE = "^print"
```

EXCLUDE_DIR

Tragen Sie hier einen regulären Ausdruck für die Verzeichnisse ein, die von der Link-Konsistenzprüfung nicht kontrolliert werden sollen. Verzeichnisse, deren Name auf das Muster passt, werden ausgelassen.

```
EXCLUDE_DIR = "test"
```

EXCLUDE_FULL_DIR

Mit dieser Direktive können Sie vollständige Pfade angeben, die von der Link-Konsistenzprüfung ausgelassen werden sollen. Nur Pfade, die exakt dem eingetragenen Wert entsprechen, werden übersprungen.

```
EXCLUDE_FULL_DIR = "/site"
```

AUTHBASIC_login

Wenn Sie das Zielsystem ganz oder teilweise mit einer Basic-Authentifizierung abgesichert haben, müssen Sie in der Konfiguration der Link-Konsistenzprüfung Authentifizierungsdaten angeben. Diese Daten nutzt die Link-Konsistenzprüfung bei der Kontrolle der Links zur Anmeldung bei dem betreffenden System. Wenn Sie diese Angaben nicht in der Konfigurationsdatei eintragen, werden sämtliche Verweise in den zugangs geschützten Bereich im Prüfbericht als defekte Links aufgeführt. In der Spalte mit dem HTTP-Fehler erscheint dann die Meldung "401 Authorisation required".

```
AUTHBASIC_login = "login"
```

AUTHBASIC_password

Tragen Sie hier das Passwort für die Anmeldung beim zugriffsgeschützten Bereich des zu prüfenden Systems ein.

```
AUTHBASIC_password = "password"
```

ROOT_PAGES

Mit dieser Direktive definieren Sie die Startseiten für die Link-Konsistenzprüfung. Von den hier angegebenen Seiten aus folgt die Link-Konsistenzprüfung allen Verweisen. Diese Einstellung ist relevant für die Kategorie "Verwaiste Objekte" im Prüfbericht.

Mehrere Seiten notieren Sie durch Leerzeichen voneinander getrennt hintereinander.

```
ROOT_PAGES = "/index.html.de /index.html.en"
```

HTTP_PROXY

Wenn der Zugriff auf das zu prüfende System über einen Proxy-Server erfolgt, müssen Sie dessen URL in dieser Direktive eintragen.

```
HTTP_PROXY = "http://proxy:3128"
```



Wichtig:

Die Link-Konsistenzprüfung wertet die folgenden Direktiven nur aus, wenn Sie auch einen HTTP-Proxy definiert haben.

HTTPS_PROXY

Notieren Sie hier den Proxy für verschlüsselte Datenübermittlung mit SSL.

```
HTTPS_PROXY = "http://proxy:3128"
```

FTP_PROXY

Mit dieser Direktive definieren Sie den Proxy für das FTP-Protokoll.

```
FTP_PROXY = "http://proxy:3128"
```

NO_PROXY

Hier geben Sie die URLs an, auf die direkt zugegriffen werden soll.

```
NO_PROXY = "http://www.mynoproxy.de"
```

3.12.1 Konfigurationsbeispiel

Im Verzeichnis site/config Ihrer Imperia-Installation finden Sie die Datei linkmanager.conf.sample. Kopieren Sie diese Datei nach linkmanager.conf und editieren Sie die Kopie nach Ihren Anforderungen, um die Link-Konsistenzprüfung schnell einsetzen zu können.

```
### Linkmanager Sample Configuration ###
<SERVER mycool.server>
  SERVERNAME = "Live-System"
  SERVER_URL = "http://mycool.server.net"

  #SETTYPE_HTML = "\..?htm.?|\..?php"
  #SETTYPE_CSS = "\.css/i"

  # a regexp, paths containing the given pattern are excluded
  #EXCLUDE_DIR = "test"
  #EXCLUDE_DIR = "backup"

  # same as EXCLUDE_DIR for files
  EXCLUDE_FILE = "^print"

  # full path to be excluded
  EXCLUDE_FULL_DIR = "/site"
  EXCLUDE_FULL_DIR = "/cgi-bin"
  EXCLUDE_FULL_DIR = "/icon"
  EXCLUDE_FULL_DIR = "/imperia"

  # define the pages from which the link consistency check is started
  ROOT_PAGES = "/index.html /index.html.de /index.html.en"

  # setting http_proxy directive is mandatory for the rest of the
  # proxy-settings to be considered
  #HTTP_PROXY = "http://www.proxy.net:8020"
  HTTPS_PROXY = "http://www.proxy.net:8021"
  FTP_PROXY = "http://www.proxy.net:8022"
  NO_PROXY = " http://www.proxy.net:8023"

</SERVER>
```

3.13 Erweiterung des Schreibtischs

Die Einstellungen des Schreibtischs von Imperia lassen sich nicht nur auf der Benutzerebene individuell beeinflussen (siehe Kapitel "Schreibtisch" im Benutzerhandbuch).

Als Administrator können Sie beliebige Schreibtischansichten definieren, die systemweit für alle Benutzer gültig sind. Dabei können Sie sowohl neue Funktionen hinzufügen als auch Standardfunktionen ausblenden, die nicht verfügbar sein sollen.

Diese Änderungen nehmen Sie über Perl-Module vor, die updatesicher im System gespeichert werden.

3.13.1 Schreibtischansichten einrichten

Für die Definition eigener updatesicherer Schreibtischansichten benötigen Sie folgende Komponenten:

Spaltendefinition

Die Konfiguration der verfügbaren Spalten nehmen Sie in einem Perl-Modul vor. Die Eigenschaften der einzelnen Spalten werden darin in Form eines Perl-Hashs festgehalten. Dazu gehört unter anderem die Angabe, welches Perl-Modul die Funktion der Schreibtisch-Spalte enthält.

Als Beispiel für die verfügbaren Konfigurationsmöglichkeiten können Sie sich auf Ihrem Produktionssystem die Datei `site/modules/core/Imperia/Desktop/DefaultColumns.pm` ansehen. Sie enthält die Konfiguration für die Standard-Schreibtisch-Spalten von Imperia.

Wenn Sie eigene Spalten hinzufügen möchten, brauchen Sie nur ein Modul mit einem Hash, der die Eigenschaften der Zusatzspalten enthält, nach diesem Muster anzulegen. Es ist auch möglich, parallel mehrere Module zu nutzen.



Wichtig:

Achten Sie bei der Verwendung mehrerer Module mit Spaltendefinitionen darauf, dass Sie eindeutige Namen für Ihre zusätzlichen Spalten vergeben.

Stellen Sie sicher, dass von Ihnen verwendete Spaltennamen nicht schon für Imperia-Standard-Schreibtisch-Spalten in Gebrauch sind. Die von Ihnen definierten Spalten ersetzen in diesem Fall die Standard-Spalten.

Die Dateien müssen auf Ihrem Produktionssystem in einem Verzeichnis unterhalb von `site/modules/core/` abgelegt werden.

Auf dem Schreibtisch sichtbar werden auf diese Weise definierte Spalten erst, wenn sie in eine Schreibtisch-Ansicht eingebunden werden. Zusätzlich müssen Sie durch zwei System-Konfigurationsvariablen die Erweiterung Ihres Schreibtischs aktivieren. Lesen Sie hierzu Abschnitt [3.13.2 Schreibtisch-Erweiterungen in das System einbinden](#) auf Seite 48.

Im Dialog Schreibtisch-Konfiguration erscheinen von Ihnen definierte Schreibtisch-Spalten ebenfalls. Sie sind in der Pull-Down-Liste zur Auswahl hinzuzufügender Spalten aufgeführt und können auf diese Weise auch in bestehende Schreibtischansichten integriert werden.

Definition einer Schreibtisch-Ansicht

Ähnlich wie die Definition zusätzlicher Schreibtischspalten geschieht auch die Einrichtung von Schreibtischansichten durch Perl-Module. Hier werden die Eigenschaften in Form eines Perl-Hashs abgelegt. Die Definition der Standard-Schreibtischansichten liegt beispielsweise auf dem Produktionssystem in der Datei `site/modules/core/Imperia/Desktop/DefaultConfig.pm`.

Legen Sie für Ihre zusätzliche Ansicht ein eigenes Modul an. Beliebig viele Module sind möglich. Sie können aber auch alle selbst entworfenen Ansichten in einem einzigen Modul speichern.



Wichtig:

Verwenden Sie für Ihre Schreibtischansichten eindeutige Namen und beachten Sie, dass diese nicht bereits vergeben sind.

Ebenso wie Spaltendefinitionen müssen Schreibtischansichten in einem Verzeichnis unterhalb von `site/modules/core/` auf dem Produktionssystem abgelegt werden. Sobald Sie ein solches Modul angelegt haben, sind die darin definierten Schreibtischansichten allen Benutzern zugänglich.

Module für Schreibtischfunktionen

Für die eigentliche Funktionalität einer Schreibtischerweiterung brauchen Sie ein separates Perl-Modul. Legen Sie es auf dem Produktionssystem im Verzeichnis `site/modules/core/Dynamic/DesktopColumn` ab.

3.13.2 Schreibtisch-Erweiterungen in das System einbinden

Zur Aktivierung eigener Schreibtischerweiterungen müssen Sie diese mit zwei Konfigurationsvariablen ins System einbinden. Die Variablen werden in der Datei `system.conf` definiert:

DESKTOP_COLUMN_MODULES

Weisen Sie dieser Variablen als Wert eine durch Semikolons getrennte Liste von den Package-Namen der Module mit Spaltendefinitionen zu.

Dazu ein Beispiel:

```
"DESKTOP_COLUMN_MODULES" = "Imperia::Desktop::myDefaultColumns;mycols::mySpecialColumns"
```

Hier werden zwei Module ins System eingebunden. Die zugehörigen Dateien sind auf dem System unter site/modules/core/Imperia/Desktop/myDefaultColumns.pm und site/modules/core/mycols/mySpecialColumns.pm abgelegt.

DESKTOP_CONFIG_MODULES

In dieser Variablen speichern Sie die Package-Namen der Perl-Module mit Ihren Schreibtischansichten. Die einzelnen Namen sind durch Semikolons getrennt zu notieren. Notieren Sie beispielsweise folgendes, um die beiden Module site/modules/core/Imperia/Desktop/myDefaultConfig.pm und site/modules/core/mycols/mySpecialConfig.pm einzubinden:

```
"DESKTOP_CONFIG_MODULES" = "Imperia::Desktop::myDefaultConfig;myconf::mySpecialConfig"
```

3.13.3 Standard-Schreibtischansichten umgehen

Der Aufbau des Imperia-Schreibtisches findet in mehreren Phasen statt. Es werden systemweite Standardinstellungen sowie Erweiterungen berücksichtigt. Die jeweiligen Einstellungen für Spalten und Schreibtischansichten werden dabei in einer festen Reihenfolge ausgewertet:

Reihenfolge bei der Auswertung von Spaltendefinitionen:

1. Standard-Schreibtisch-Spalten (Modul *Imperia::Desktop::DefaultColumns*).
2. Dateien mit der Endung *.col, die unter site/config/desktop abgespeichert sind. Hierbei handelt es sich um Spaltendefinitionen, die von einem Benutzer über die Imperia-Oberfläche angelegt und entsprechend kopiert wurden, damit Sie allen Nutzern des Systems zur Verfügung stehen (siehe Kapitel "Schreibtisch" im Benutzerhandbuch).
3. Spaltendefinitionen, die in der Konfigurationsvariablen DESKTOP_COLUMN_MODULES aufgeführt sind.
4. Benutzerspezifische Einstellungen.

Reihenfolge bei der Auswertung von Schreibtischansichten:

1. Standard-Ansichten (Modul *Imperia::Desktop::DefaultConfig*).
2. Erweiterungen, die in der Variablen DESKTOP_CONFIG_MODULES eingetragen sind.
3. Benutzerspezifische Einstellungen.

Sowohl bei Spaltendefinitionen als auch bei Schreibtischansichten gilt, dass später ausgewertete Einstellungen etwaige bereits bestehende überschreiben.

Auf diese Weise können Sie beispielsweise die Anzeige von bestimmten Spalten auch in den Standard-Ansichten unterdrücken. Diese Änderung gilt dann für alle Benutzer.

Definieren Sie dazu eine eigene Schreibtisch-Ansicht mit dem Namen der betreffenden Standard-Ansicht und lassen Sie die Spalte aus, die Sie ausblenden wollen. Dazu ein Beispiel:

Sie möchten die Anzeige des Dokumententitels in der Default-Ansicht des Schreibtisches unterbinden. Die Definition der Ansicht steht im Modul site/modules/core/Imperia/Desktop/DefaultConfig.pm:

```
# Ausschnitt aus dem Package Imperia::Desktop::DefaultConfig

Default => {
    columns => [
```

```

        [ qw (grid_status) ],
        [ qw (edit) ],
        [ qw (title) ],
        [ qw (category) ],
        [ qw (modified) ],
        [ qw (next_step) ],
    ],
    sort => [ qw (modified created node_id) ],
},

```

Natürlich können Sie auf die selbe Weise auch Erweiterungen der Standard-Ansichten vornehmen.

Um die Spalte auszublenden, die den Dokumententitel anzeigt, legen Sie ein eigenes Modul an und definieren darin die Default-Ansicht entsprechend:

```

# Ausschnitt aus dem Package Imperia::Desktop::myDefaultConfig

Default => {
    columns => [
        [ qw (grid_status) ],
        [ qw (edit) ],
        [ qw (category) ],
        [ qw (modified) ],
        [ qw (next_step) ],
    ],
    sort => [ qw (modified created node_id) ],
},

```

3.14 Skinning der Flexmodul-Steuerelemente

Den Quellcode der Flexmodul-Steuerelemente können Sie leicht abändern oder komplett ersetzen, indem Sie den Imperia Standard-Code durch selbst erstellte Skins ersetzen.

Flexmodul-Skins ähneln grundsätzlich Codeincludes. Legen Sie jeweils für die Flexmodul-Kontrollleiste und die Steuerelemente der Flexinstanzen eine eigene Datei mit dem jeweiligen Quellcode an und speichern Sie diese im Verzeichnis `site/skin/flex`, in dem Sie auch Beispieldateien für Flexmodul-Skins finden. Die Namenskonvention für Skindateien lautet wie folgt:

```

MYNAME.control.htm => Skin-Datei für die Flexmodul-Kontrollleiste
MYNAME.instance.htm => Skin-Datei für die Steuerelemente der Flexinstanzen

```



Wichtig:

MYNAME muss für beide Dateien gleich lauten, damit die Einbindung über den Flexmodulauftrag im Template funktioniert.

Die Einbindung Ihrer Flexmodul-Skins können Sie entweder dokumentenbasiert durch einen Parameter im Flexmodulauftrag oder durch zwei Variablen zur Systemkonfiguration vornehmen. Für Einzelheiten zum Flexmodul-Skinning auf Dokumentenbasis lesen Sie bitte den Abschnitt **SKIN** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

3.14.1 FLEXMODULE-SKIN

Mit dieser Variable geben Sie den Dateinamen für Ihr Flexmodul-Skin an, mit dem Sie systemweit das tabellenbasierte Standard-Layout der Flexmodul-Steuerelemente überschreiben möchten. Geben Sie nur den Dateinamen an, wenn Sie die Datei im Standard-Verzeichnis für Flexmodul-Skins, `site/skins/flex` ablegen möchten. Alternativ können Sie eine Pfadangabe ausgehend von der Document-Root Ihres Entwicklungssystems notieren.

3.14.2 FLEXINSTANCE-SKIN

Wie FLEXMODULE-SKIN, nur für die Steuerleiste der Flex-Instanzen.

3.15 Änderungsverfolgung

Die administrativen Einstellungen für die Änderungsverfolgung beschränken sich auf die Bestimmung der Farben, mit denen die Änderungen dargestellt werden. Diese Farben werden in einem gesonderten Stylesheet definiert.

Folgende Änderungen können angezeigt werden:

- hinzugefügter Text (im Standard grün und unterstrichen dargestellt).
- gelöschter Text (im Standard rot und durchgestrichen dargestellt).
- hinzugefügte Tabellen, Tabellenzellen und Anker (im Standard hellgrün dargestellt).
- gelöschte Tabellen, Tabellenzellen und Anker (im Standard orange dargestellt).
- geänderte Bilder (im Standard mit einem durchgehenden orangefarbenen Rand dargestellt).
- eingefügte Zeilenumbrüche (im Standard wird der umbrochene Text in der neuen Zeile grau dargestellt).
- Verweise, auf den ein IFrame zeigt (im Standard wird der IFrame mit einem durchgehenden orangefarbenen Rand dargestellt).

Es bestehen zwei Möglichkeiten, die Farben zu ändern, die für die Darstellung der Änderungen verwendet werden:

- für das gesamte System, in dem die Standardeinstellungen im entsprechenden Stylesheet geändert werden.
- über Style-Angaben im Template.

3.15.1 Änderung der Standardfarben

Werden die Standardfarben der Änderungsverfolgung im Stylesheet geändert, gelten diese Änderungen für alle Dokumente im gesamten Produktionssystem.

Das für die Farben der Änderungsverfolgung verantwortliche Stylesheet liegt im Verzeichnis <Document-Root des Produktionssystems>/imperia/skin/html/css unter dem Namen diff.css.

Es stehen folgende CSS-Formate zur Verfügung:

imperiaPlusSpan

Bestimmt das Aussehen von hinzugefügtem Text.

imperiaMinusSpan

Bestimmt das Aussehen von gelöschtem Text.

imperiaPlusSpan table, td, a

Bestimmt das Aussehen von hinzugefügten Tabellen, Tabellenzellen und Ankern.

imperiaMinusSpan table, td, a

Bestimmt das Aussehen von gelöschten Tabellen, Tabellenzellen und Ankern.

imperiaPlusSpan img iframe

Bestimmt das Aussehen von hinzugefügten Bildern und IFrames.

imperiaMinusSpan img iframe

Bestimmt das Aussehen von gelöschten Bildern und IFrames.

imperiaNewlineChangesSpan

Bestimmt das Aussehen von umbrochenem Text.

imperialImageChanged

Bestimmt das Aussehen von geänderten Bildern.

imperialFrameChanged

Bestimmt das Aussehen von geänderten IFrames.

Eine weitere Möglichkeit, die Standardfarben zu ändern, besteht darin, innerhalb des Templates Meta-Variablen zu erzeugen, die andere Farbwerte enthalten. Lesen Sie hierzu das Kapitel **Änderungsverfolgung** im Programmierhandbuch.

3.16 Reparsen umfangreicher Sites

Zum Reparsen einer umfangreichen Website steht das Skript `site/bin/xreparse.pl` zur Verfügung. Eine Durchführung über die Imperia-Oberfläche ist nicht vorgesehen. Dieses Skript ist im Gegensatz zu der entsprechenden über die Benutzeroberfläche ausführbaren Funktionen auch für große Datenmengen geeignet.

Mit Hilfe des Skripts können alle Dokumente, die sich in der Freischaltliste oder auf dem Zielsystem befinden, bearbeitet werden. Es wird vorausgesetzt, dass sich keines dieser Dokumente im Workflow befindet. Wir empfehlen weiterhin, den Workflow für alle Dokumente zu beenden, bevor dieses Skript ausgeführt wird.

`xreparse.pl` ignoriert Dokumente des Media Asset Management, es sei denn, dies wird mit dem Parameter `-m` erzwungen.



Hinweis:

Die Meta-Variable `publish_date` wird in den Dokumenten, die auf den mit dem Parameter `-p` gesetzten Veröffentlichungsbedingungen passen, auf den Wert `2001-01-01 01:01` gesetzt.

3.16.1 Optionen

Das Skript wird auf der Kommandozeile mit folgenden Optionen gesteuert:

-h

Zeigt eine kurze Hilfe zum Skript.

-k

Behält das Veröffentlichungsdatum beim Reparsen bei ("keep").

-p Argument

Dieser Parameter kontrolliert das Veröffentlichungsverhalten der reparsten Dokumente. Als Argumente sind `none` und `all` erlaubt.

Wird das Argument `none` gesetzt, werden alle Dokumente nach dem Reparsen in die Freischaltliste eingetragen, werden aber nicht freigeschaltet.

Wird das Argument `all` gesetzt, werden alle Dokumente, die sich bereits auf dem Zielsystem befinden, nach dem Reparsen erneut freigeschaltet, wobei auch andere Dokumente, die sich in der Freischaltliste befinden, mit freigeschaltet werden.

Wird die Option `-p` nicht verwendet, werden alle reparsten Dokumente, die sich bereits auf dem Zielsystem aber nicht in der Freischaltliste befinden, wieder freigeschaltet.

-l

Dieser Parameter reparst Dokumente und fügt sie anschließend der Freischaltliste hinzu oder veröffentlicht die Dokumente, wenn sie sich schon auf dem Zielsystem befinden.

-w

Die Dokumente werden nicht reparst, sondern re-importiert und in den Workflow geleitet. Zusätzlich setzt dieser Parameter die Umgebungsvariable `__IMPERIA_XREPARSE` auf einen wahren Wert. Die Variable können Sie dazu verwenden, reparste Dokumente im Workflow mit Hilfe eines IfElse-Plug-Ins in einen anderen Workflow-Zweig zu lenken. Die Bedingung dieses IfElse-Plug-Ins sollte wie folgt aussehen:

```
"<!--ENV:__IMPERIA_XREPARSE-->" AND (NOT "<!--ENV:GATEWAY_INTERFACE-->")
```

-f

Diese Option erzwingt die Ausführung des Skripts ohne weitere Usereingabe. Wird die Option nicht verwendet, wird eine Empfehlung, `STDOUT` in eine Log-Datei umzulenken, angezeigt.

-e

Wird diese Option in Zusammenhang mit `-w` verwendet, wird der erste Workflow-Schritt eines jeden reparsten Dokuments ausgeführt. Das Standard-Verhalten ist, den ersten Workflow-Schritt zu überspringen.

-n

Wird diese Option verwendet, interpretiert das Skript eine eventuell vorhandene Liste von Argumenten als Liste von NodeIDs, während standardmäßig eine Liste von Argumenten als Liste von Template-Namen interpretiert wird. Dadurch ist es möglich, das Reparsen auf eine Rubrik und deren Unterrubriken zu beschränken.

-m

Diese Option erzwingt das Reparsen auch für die Dokumente der MAM. Da deren Reparsen selten sinnvoll ist, werden MAM-Dokumente im Allgemeinen von `xreparse.pl` nicht bearbeitet.

3.16.2 Argumente

Standardmäßig interpretiert das Skript eine Liste von Argumenten als Liste von Template-Namen, wobei das führende *template* und das *htms*-Suffix nicht angegeben werden. Wird die Option `-n` verwendet, wird die Liste der Argumente jedoch als Liste von Rubriken-NodeIDs interpretiert.

Das Reparsen wird auf die Dokumente beschränkt, deren Templates auf die angegebenen Namen passen bzw. deren NodeIDs mit einer der angegebenen Rubriken-NodeIDs beginnen.

3.16.3 Beispiel

Dieses Beispiel eignet sich für einen Workflow, dessen erster Schritt interaktiv, alle folgenden Schritte jedoch nicht interaktiv sind. Dadurch können die reparsten Dokumente den Workflow durchlaufen, ohne dass ein Usereingriff notwendig ist.

```
./xreparse.pl -w -p all -f -e Service Produkt
```

Es werden alle Dokumente reparst, die mit dem Template `templateService.htms` oder dem Template `templateProdukt.htms` erzeugt wurden. Beim Reparsen wird die Umgebungsvariable gesetzt. Es bedarf keiner weiteren Benutzereingabe. Nach dem Reparsen wird der erste Workflow-Schritt ausgeführt. Die reparsten Dokumente werden freigeschaltet.

3.17 Das interne Mail-System

Das Imperia-eigene Mailsystem bedarf keiner Administration. Es kann aber deaktiviert werden, indem über das Hauptmenü - Menüs - Skripte - Filter:all die Skript-Zugriffsrechte auf das Skript `site_mailer.pl` entfernt werden.



Hinweis:

Für Benutzer mit Superuser-Rechten hat diese Einstellung keinen Effekt, da Superuser immer alle Rechte haben.

Kapitel 4. Struktur

Unter dem Hauptmenüpunkt **Struktur** finden Sie Funktionen zur Verwaltung von Rubriken, Metadateien, Templates, Flexmodulen, Codeincludes, Slots-Modulen, Verschieben von Assets und den Template Reparser. Informationen zum Erstellen bzw. Programmieren dieser Dateien finden jeweils in einem eigenen Kapitel. In diesem Kapitel sind lediglich die Verwaltung und Handhabung dieser speziellen Dateien über die Imperia-Oberfläche beschrieben.

Weitere Themen sind Dynamische Module, der Template-Reparser und der Abgleich der Systemdateien zwischen Produktions- und Zielsystem.

4.1 Rubriken

Das Rubriken-Management erreichen Sie durch Auswahl des Menüpunktes **Hauptmenü -> Struktur -> Rubriken**. Sie gelangen zu folgendem Dialog (hier mit Beispielrubriken):

Rubriken bearbeiten		Bearbeiten	Rechte	Neu	Löschen	Runter	Hoch
☰	Ionensturm Digitalbau GmbH Internet	▼	☒	+	X	▼	
	└ Startseite (deutsch)	▼	☒	+	X	▼	
	└ Startseite (englisch)	▼	☒	+	X	▼	↑
	└ Das Unternehmen	▼	☒	+	X	▼	↑
	└ Aktuelles	▼	☒	+	X	▼	↑
	└ Stellenangebote	▼	☒	+	X	▼	↑
	└ Glossar	▼	☒	+	X	▼	↑
	└ Textbausteine	▼	☒	+	X	▼	↑
	└ Presse	▼	☒	+	X		
	└ Formulare	▼	☒	+	X	▼	↑
	└ Miniweb Ansprechpartner	▼	☒	+	X	▼	↑
	└ Miniweb Standorte	▼	☒	+	X	▼	↑
	└ Gästebuch	▼	☒	+	X		
☰	Ionensturm Digitalbau GmbH Intranet	▼	☒	+	X	▼	↑
	└ Navigation Internet	▼	☒	+	X	▼	↑
	└ Navigation Intranet	▼	☒	+	X	▼	↑
	└ Portaldemo	▼	☒	+	X		
Neue Hauptrubrik							

Abb. 4.1: Dialog Rubriken bearbeiten

Hier finden Sie Funktionen zum Einrichten, Bearbeiten und Löschen von Rubriken und zum Bearbeiten der jeweiligen Zugriffsrechte. Bestehende Rubriken werden in einer Baumstruktur angezeigt.

Rubriken müssen nicht zwangsläufig Dokumente enthalten. Sie können durch Schachtelung auch die thematische Struktur Ihres Projektes abbilden. Einige Rubriken enthalten dabei dann lediglich weitere Unterrubriken. Lesen Sie hierzu auch Abschnitt 6.1.1 **Rubriken-Informationen angeben** auf Seite 86.

Die Einstellung, ob in einer Rubrik Dokumente angelegt werden können oder nicht, kann jederzeit geändert werden. Dadurch kann man eine Rubrik "einfrieren" und vermeiden, dass weitere Dokumente angelegt werden.

Der hier erzeugte Rubrikenbaum wird auch im Archiv, beim Erstellen neuer Dokumente und beim Bearbeiten der Rubriken verwendet. Für jede dieser Ansichten wird jedoch getrennt abgespeichert, welche Rubriken aufgeklappt sind und welche nicht.

Das Rubriken-Management steht nur Rollen mit entsprechender Zugriffsberechtigung zur Verfügung. Wenn Sie den Menüpunkt **Rubriken** nicht im Hauptmenü sehen, hat Ihre derzeitige Rolle keinen Zugriff auf diese Funktion. Nehmen Sie eine Rolle mit ausreichender Berechtigung an oder wenden Sie sich an den Imperia-Systemadministrator (**Superuser**).

4.1.1 Neue Rubrik anlegen

Neue Rubriken werden entweder als Hauptrubrik, also als ein Wurzelement des Rubrikenbaums, oder als Unterrubrik einer Hauptrubrik angelegt. Eine neue Hauptrubrik wird über den Link **Neue Hauptrubrik** in der Titelleiste des Dialogs erzeugt.

Wenn Sie eine Unterrubrik anlegen wollen, klicken Sie in der Zeile der entsprechenden Hauptrubrik auf folgenden Button:



Halten Sie dabei gleichzeitig die Taste **STRG** gedrückt, um alle Unterknoten zu öffnen bzw. zu schließen. Neue Unterrubriken werden immer unterhalb der Zeile der Hauptrubrik angelegt. Nach dem Betätigen des Buttons wird der Dialog **Neue Rubrik anlegen** geöffnet.

Dieser Dialog ist in vier Abschnitte unterteilt, die in Kapitel 6.1 beschrieben werden. Eine genaue Beschreibung der einzelnen Abschnitte finden Sie in Abschnitt 6.1 **Rubriken-Eigenschaften bestimmen** auf Seite 85.

Eingaben werden über den Button **Okay** am unteren Rand des Dialogs gespeichert.

4.1.2 Rubriken bearbeiten

In der Spalte **Bearbeiten** des Dialogs **Rubriken bearbeiten** befindet sich in jeder Zeile ein Button, mit dem die entsprechende Rubrik zum Bearbeiten geöffnet werden kann:



Der Button öffnet den Dialog **Rubrik bearbeiten**, in dem Sie die vorhandenen Einstellungen in den entsprechenden Abschnitten ändern können. Zur Speicherung der Änderungen klicken Sie auf den Button **Speichern** am unteren Rand der Imperia-Oberfläche.

4.1.2.1 Rubriken-Metainformationen

Zusätzlich zu den Einstellungen, die Sie bei der Erstellung der Rubriken vornehmen, können Sie bei der Bearbeitung bereits bestehender Rubriken Metainformationen mit der Rubrik verknüpfen. Klicken Sie hierzu auf den Link **Bearbeiten**. Sie gelangen zu einem Meta-Viewer, in dem Sie Metainformationen hinzufügen, entfernen oder ändern können.



Hinweis:

Diese Funktion steht nur Benutzern mit Schreibrecht auf die Rubrik zur Verfügung.

Alternativ können Sie den Dialog zum Bearbeiten der Rubrik-Metainformationen auch direkt aus der Rubriken-Verwaltung aufrufen. Klicken Sie hierzu auf das folgende Icon:



Rubrik-Metainformationen vererben sich automatisch beim Anlegen von Unterrubriken. Das heißt, die Metainformationen sind dort dann mit den Metainformationen der übergeordneten Rubrik vorbelegt.



Hinweis:

Der Zugriff auf Rubrik-Metainformationen in Dokumenten lässt sich etwas einfacher mit dem Workflow-Plug-In Cat2DocMeta umsetzen. Lesen Sie hierzu Abschnitt 8.5.8



Cat2DocMeta auf Seite 113.

Wozu brauche ich diese Meta-Informationen? Statt das DESCRIPTION-Feld der Rubriken mit Werten zu füllen, um zum Beispiel Variablen für die Navigationserzeugung zu speichern, kann man hier sehr umfangreich eigene Variablen an Rubriken hängen. Auf diese kann man über Perl sehr einfach zugreifen: wo immer der Nodeinfo-Hash der Rubrik zurückkommt oder gesetzt wird, gibt es unter dem Key `META_INFO` ein `Meta::Info`-Objekt. Die Klasse `Category::Category` wurde um `getMetaInfo()` und `setMetaInfo()` erweitert. Wer bei Verwendung der Filesystem-Datenhaltung die Meta-Informationen an Imperia vorbei mit `stdump/stfreeze` lesen bzw. manipulieren will, sollte beachten, dass dort nicht das `Meta::Info`-Objekt, sondern die blanke Hash-Referenz, bei der es sich um kein Objekt handelt, abgespeichert wird. In der Datenbank dagegen liegen die Meta-Informationen in der neuen Tabelle `category_meta`.

4.1.2.2 Rubriken-Metainformationen bearbeiten

Enthält das Metafeld `__imperia_category_template` einer Rubrik einen gültigen Templatennamen, wird zum Editieren stattdessen dieses Template verwendet. Falls das Metafeld nicht vorhanden ist, wird die System-Conf-Variable `CATEGORY_TEMPLATE` herangezogen und ansonsten das Default-Template verwendet. Mit diesem Mechanismus können Sie also die Verwendung ganz bestimmter Metainformationen erreichen.

Im Folgenden wird die Verwendung des Imperia-Fallback-Templates erklärt.

Meta field	Values	
__imperia_ac_enabled	0	Move up Edit Clone Delete Move down Ty + X
__imperia_node_id	/1	
<Add Field>		
<Recursive Meta Actions>		

Um eine Metainformation einzugeben, tragen Sie zuerst in das Eingabefeld den gewünschten Variablennamen ein, in unserem Beispiel ist das „linguas“. Klicken Sie dann auf den Link „<Feld hinzufügen>“. Nach dem Klick erscheint der Variablenname in der Liste mit einem leeren Eingabefeld. Hier kann jetzt ein Wert eingegeben werden, in unserem Fall „de“. Sie können die Eingabe jetzt durch einen Klick auf „Speichern“ beenden. Wenn Sie mehrere Werte als Array speichern wollen, klicken Sie auf das Plus-Icon und ein neues leeres Eingabefeld erscheint.

Meta field	Values	
__imperia_ac_enabled	0	Move up Edit Clone Delete Move down Ty + X
__imperia_node_id	/1	
linguas	de	Move up Edit Clone Delete Move down Ty + X
<Add Field>		
<Recursive Meta Actions>		

4.1.2.3 Zugriff auf die Rubriken-Metainformationen

Der Zugriff auf die Metainformationen erfolgt wie bei jedem anderen Rubrikenfeld. Ein eventueller `[index]` in Klammern liest aus einem Array das entsprechende Feld aus.

Alternativ können Sie einem Meta-Feld auch `META_INFO_` voranstellen, um normale Rubrikenfelder von Metafeldern zu unterscheiden. Das ist dann nützlich, wenn Sie einen Namen verwenden möchten, der bereits als normales Feld in Rubriken verwendet wird. Das wird selten notwendig sein, kann aber in Ausnahmefällen helfen.

Dazu ein Beispiel:

Angenommen, die Meta-Informationen der Rubrik mit Namen "Test", der ID "/123/456/789" und dem Directory "/test" sind folgendermaßen befüllt:

```
example | "Ein Beispieltext"
DIRECTORY | "Meta-Info, nicht das echte Directory"
list | "eins" | "zwei" | "drei"
```

In einem (Dokumenten!)-Template verwenden Sie jetzt:

```
category: <!--SECTION:NAME:0--><br />
example(M) : <!--SECTION:META_INFO_example:0--><br />
example: <!--SECTION:example:0--><br />
DIRECTORY: <!--SECTION:DIRECTORY:0--><br />
DIRECTORY(M) : <!--SECTION:META_INFO_DIRECTORY:0--><br />
List: <!--SECTION:list[0]:0--><br />
List: <!--SECTION:list[1]:0--><br />
List: <!--SECTION:list[2]:0--><br />
List: <!--SECTION:list:0--><br />
Rubriken-ID: <!--SECTION:__imperia_node_id:0--><br />
```

Dann macht der Template-Prozessor daraus:

```
category: Test
example(M) : Ein Beispieltext
example: Ein Beispieltext
DIRECTORY: /test
DIRECTORY(M) : Meta-Info, nicht das echte Directory
List: eins
List: zwei
List: drei
List: eins
Rubriken-ID: /123/456/789
```

Beispiel 4.1. Zugriff auf Meta-Informationen von Rubriken

4.1.2.4 Verfügbare Sprachen für mehrsprachige Dokumente festlegen

Ein weiteres Beispiel für die Verwendung von Rubrik-Metainformationen ist das Metafeld `linguas`. Es speichert die gewünschten Sprachen in einem Array. Jedes Element dieses Arrays besteht aus einem ISO 639 Sprach-Code für eine Sprache (de, en, fr etc.). Imperia baut aus dem Inhalt dieser Metavariablen die Sprachauswahl im Meta-Edit und im Bearbeiten-Schritt des Workflows auf.

4.1.3 Rubrikenrechte einstellen

Für jede Rubrik können für jede Rolle im System verschiedene Zugriffsrechte eingestellt werden. Diese Rechte bestimmen die Funktionen, die einem User für die Bearbeitung von Dokumenten in dieser Rubrik zur Verfügung stehen. Mögliche Rechte sind:

- Lesen
- Dokumente erzeugen
- Schreiben
- Löschen

Die Rechte einer Rolle in einer Rubrik lassen sich über die Funktion **Rubrik bearbeiten** (siehe Abschnitt 4.1.2 **Rubriken bearbeiten** auf Seite 56) und im Dialog **Rubriken-ACL-Verwaltung** einstellen. Diesen Dialog öffnen Sie für jede Rubrik über den Button **Zugriffsrechte** im Dialog **Rubriken bearbeiten**. Einer der Hauptunterschiede zu der im vorigen Abschnitt beschriebenen Bearbeitung der Rechte ist die Möglichkeit, in der ACL-Verwaltung Zugriffsrechte rekursiv einzustellen. Sie können hier also gleich für ganze Teilrubrikenbäume die Berechtigungen einstellen.

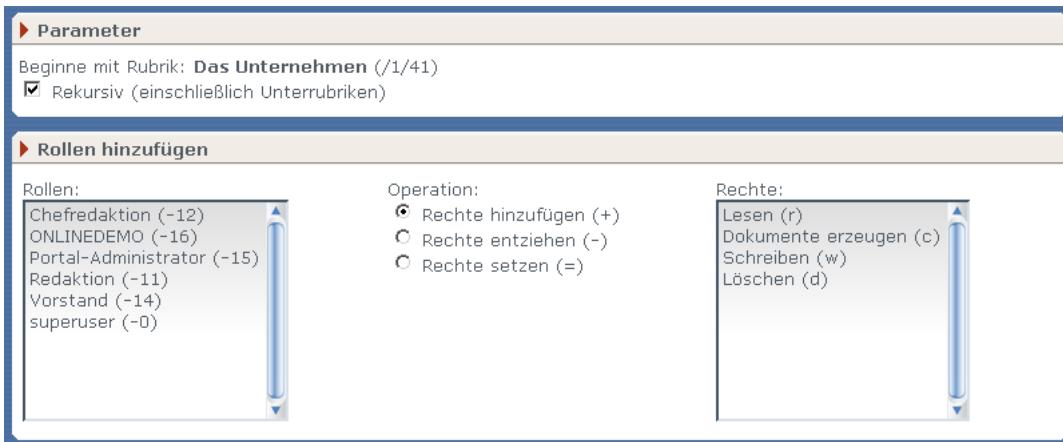


Abb. 4.2: Rubriken-ACL-Verwaltung

Der Dialog ist in zwei Abschnitte unterteilt:

Im ersten Abschnitt **Parameter** legen Sie fest, ob die Änderung der Rechte rekursiv, also auch für enthaltene Unterrubriken, gelten soll.

Im zweiten Abschnitt **Rollen hinzufügen** weisen Sie die Rechte einzelnen oder mehreren Rollen zu. Die Liste **Rollen** enthält alle verfügbaren Rollen. Markieren Sie in der Liste die Rolle(n), die Sie modifizieren wollen. Für eine Mehrfachauswahl halten Sie gleichzeitig die **STRG**-Taste gedrückt.

Ihnen stehen die folgenden Optionen zur Auswahl:

Rechte hinzufügen (+)

Diese Funktion fügt die ausgewählten Rechte den bereits bestehenden Rechten der ausgewählten Rollen hinzu.

Rechte entziehen (-)

Hiermit werden die markierten Rechte den markierten Rollen entzogen.

Rechte setzen (=)

Diese Funktion setzt die Rechte der ausgewählten Rollen auf die markierten Rechte.



Wichtig:

Nicht markierte Rechte werden den Rollen entzogen.

In der Liste **Rechte** stehen die Rechte zur Auswahl, die mit der eingestellten Aktion für die markierten Rollen modifiziert werden sollen.

Der Button **Okay** führt die eingestellten Änderungen der Zugriffsrechte aus, der Button **Schließen** schließt den Dialog, ohne Änderungen zu berücksichtigen.

4.1.4 Rubrikenreihenfolge ändern

Die Reihenfolge der Einträge im Rubrikenbaum lässt sich verändern. Änderungen wirken sich dabei auch auf etwaige Unterrubriken aus. Zur Änderung der Rubrikreihenfolge verschieben Sie die gewünschte Rubrik zeilenweise über die beiden Buttons in den Spalten **Runter** und **Hoch**.

- verschiebt die Rubrik um eine Zeile nach unten.
- verschiebt die Rubrik um eine Zeile nach oben.

Rubriken lassen sich nur auf der Ebene verschieben, auf der Sie sie angelegt haben. Es ist nicht möglich, die Rubrikenhierarchie zu ändern. Sie können mit dieser Funktion also weder eine Hauptrubrik zur Unterrubrik einer anderen machen, noch eine Unterrubrik in eine Hauptrubrik umwandeln. Um dies zu realisieren, stehen die Adminskripte zur Verfügung.

4.1.5 Rubriken löschen

Das Löschen einer Rubrik geschieht über den Button **Rubrik löschen** in der entsprechenden Zeile:



Die ausgewählte Rubrik wird gelöscht.

Sie können nur Rubriken löschen, die keine Dokumente oder Unterrubriken enthalten. Versehentliches Löschen von gefüllten Rubriken wird durch eine Fehlermeldung verhindert.

4.2 Metadateien

Über das Metadatei-Management werden alle Metadateien gepflegt. Über einen Dialog können Metadateien bearbeitet, gelöscht, neu erzeugt und aus dem Dateisystem in das Imperia-System geladen werden.

Ferner können Rechte für jede Metadatei gesetzt oder entzogen werden. Hierzu gehören die Rechte

- Schreiben,
- Ändern,
- Lesen und
- Löschen.

Zum Ändern von Dateirechten lesen Sie bitte Abschnitt 4.11 **Dateirechte in Imperia verwalten** auf Seite 75.

Um das Metadatei-Management zu öffnen, wählen Sie Wählen Sie **Hauptmenü -> Struktur -> Metadatei**. Folgender Dialog öffnet sich daraufhin:

	Bearbeiten	Rechte	Löschen
Nachrichten: Dokumenten-Informationen			
erweiterte Dokumenten-Informationen			
Betriebsrat			
Work Council			
Le comité d'entreprise informe			
CDU – Begleitende Informationen			
Wörterbuch			
Veranstaltungen (DE)			
Formular: Informationen			
Templateauswahl (DE)			
Hauptseite (DE)			
Homepage (EN)			
<hr/>			
[+] Rechte			
Neue Metadatei anlegen Metadatei hochladen			

Abb. 4.3: Dialog Metadateien bearbeiten

Der Dialog ist in drei Abschnitte gegliedert:

Dateiliste

Diese Liste enthält alle eingerichteten Metadateien. Zusätzlich befinden sich hier Buttons für die Funktionen **Bearbeiten**, **Zugriffsrechte** und **Löschen**. Der Button **Bearbeiten** öffnet den internen Editor, in dem Metadateien editiert werden. Mit Hilfe des Buttons **Zugriffsrechte** wird ein Dialog geöffnet, in dem Leserechte und Rechte zum Ändern für einzelne Metadateien gesetzt werden. Über den Button **Löschen** wird die entsprechende Metadatei im System und im Dateisystem gelöscht.

Rechteverwaltung

In diesem Abschnitt, der beim Laden des Metadatei-Managements geschlossen ist und sich erst nach einem Klick auf das Pluszeichen komplett öffnet, werden Schreib-, Lese- und Löschrechte für mehrere Metadateien gesetzt. Lesen Sie hierzu Abschnitt 4.11 **Dateirechte in Imperia verwalten** auf Seite 75.

Links

Unterhalb der Rechteverwaltung befinden sich die Links **Neue Metadatei anlegen** und **Metadatei hochladen**. Lesen Sie zu diesen Funktionen Abschnitt 4.2.1 **Neue Metadatei anlegen** auf Seite 61 bzw. Abschnitt 4.2.2 **Metadatei hochladen** auf Seite 61.

4.2.1 Neue Metadatei anlegen

Um eine neue Metadatei anzulegen, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 4.2 **Metadateien** auf Seite 60 beschrieben.
2. Klicken Sie auf den Link **Neue Metadatei anlegen**. Folgender Dialog öffnet sich:

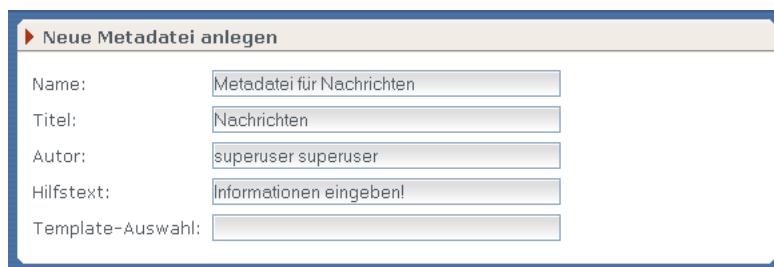


Abb. 4.4: Dialog Neue Metadatei anlegen

Tragen Sie in die einzelnen Felder die entsprechenden Daten ein. Imperia generiert anhand dieser Daten eine neue Metadatei und legt diese im Verzeichnis `/site/metafiles` ab.

Den Hilfertext bekommen die Benutzer im Meta-Edit-Schritt angezeigt. Im Quelltext der Metadatei finden Sie ihn in der Zuweisung `HELPTEXT = "..."`.

Das Feld **Template-Auswahl** ermöglicht Ihnen, den Benutzern die Wahl zu lassen, selbst das Template aus einem Pool von Templates für ihr Dokument zu bestimmen. Tragen Sie eine kommaseparierte Liste mit den Namen der Templates, die auswählbar sein sollen, ohne Dateiendungen in das Feld ein. Im Quellcode der Metadatei finden Sie diese Liste in der Zuweisung `TEMPLATESELECT "template1, template2, ..."`. Wenn Sie keine Auswahlmöglichkeit für Templates benötigen, lassen Sie das Feld leer. Dann wird automatisch das Template aus den Rubrikeinstellungen verwendet. Weitere Information hierzu finden Sie im Abschnitt **Syntax-Referenz** im Kapitel **Metadateien** des Programmierhandbuchs.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

Die neue Datei wird in die Dateiliste eingetragen und kann von dort über den Button **Bearbeiten** geöffnet und editiert werden (siehe Abschnitt 4.2.3 **Metadatei bearbeiten** auf Seite 62).

4.2.2 Metadatei hochladen

Metadateien können Sie auch außerhalb von Imperia mit einem beliebigen Texteditor erstellen. Eine eingehende Beschreibung der Syntax und des Aufbaus finden Sie im Kapitel **Metadateien** des Programmierhandbuchs.



Hinweis:

Beachten Sie, dass Metadateien UTF-8-codiert sein müssen (ohne BOM).

Um eine solche Datei in das System einzubinden, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 4.2 **Metadateien** auf Seite 60 beschrieben.
2. Klicken Sie unterhalb der Rechteverwaltung auf den Link **Metadatei hochladen**.

Es wird folgender Dialog geöffnet:

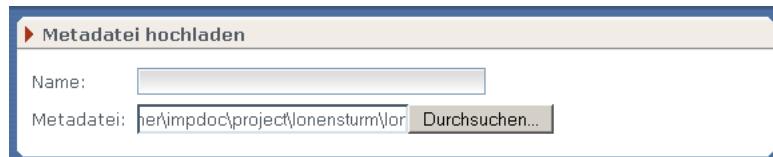


Abb. 4.5: Dialog Metadatei hochladen

3. Tragen Sie in die Felder die entsprechenden Daten ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie eine Metadatei im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis `/site/metafiles` kopiert und steht anschließend im System zur Verfügung.

4.2.3 Metadatei bearbeiten

Metadateien können im internen Editor oder einem anderen Text-Editor bearbeitet werden. Alle Metadateien sind im Verzeichnis `/site/metafiles` gespeichert. Um eine Metadatei im internen Editor zu editieren, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 4.2 **Metadateien** auf Seite 60 beschrieben.
2. Klicken Sie auf den Button **Bearbeiten** neben dem entsprechenden Metadatei-Eintrag.
In einem weiteren Fenster wird die ausgewählte Metadatei im internen Editor geöffnet.

Die Bedienung des internen Editors wird im Kapitel **Der interne Editor** im Userhandbuch beschrieben.



Tipp

Der interne Editor ist auch eine einfache Methode zur "Konvertierung" von Metadateien, die nicht im UTF-8 Zeichensatz erstellt worden sind. Kopieren Sie den Inhalt der Datei in die Zwischenablage und von dort aus in das Editorfenster. Imperia speichert die Datei automatisch im Unicode-Zeichensatz ab.

4.2.4 Metadatei löschen

Um eine Metadatei zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das Metadatei-Management wie unter Abschnitt 4.2 **Metadateien** auf Seite 60 beschrieben.
2. Klicken Sie auf den Button **Löschen** neben dem entsprechenden Eintrag in der Dateiliste.

Es erscheint eine Sicherheitsabfrage, die Sie mit **Okay** bestätigen müssen. Anschließend wird die Metadatei aus Imperia und aus dem Verzeichnis `/site/metafiles` gelöscht. Der Button **Zurück** bricht die Aktion ab.

4.3 Templates

Mit Hilfe des Template-Managements werden alle vorhandenen Templates gepflegt. Über einen Dialog können Templates bearbeitet, gelöscht, neu erzeugt und aus dem Dateisystem in das Imperia-System geladen werden.

Ferner können Rechte für einzelne oder gleichzeitig für mehrere Templates gesetzt bzw. entzogen werden. Im Einzelnen sind das folgende Rechte:

- Schreiben,
- Ändern,
- Lesen und

- Löschen.

Zum Ändern von Dateirechten lesen Sie bitte Abschnitt 4.11 **Dateirechte in Imperia verwalten** auf Seite 75.

Um das Template-Management zu öffnen, wählen Sie **Hauptmenü -> Struktur -> Templates**.

Sie gelangen zu folgendem Dialog:

▶ Templates bearbeiten					
	Beschreibung	Bearbeiten	Rechte	Löschen	
DHTML-Mini-Demo	DHTML-Mini-Demo	tx	tx	x	
Dictionary	Wörterbuch	tx	tx	x	
Import	Import	tx	tx	x	
Index	Indexseite mit Teasern	tx	tx	x	
Kantinenplan	Kantinenplan	tx	tx	x	
Navigation	Navigation	tx	tx	x	
NavigationCI		tx	tx	x	
Navigation_perso	Navigation Intranet(P)	tx	tx	x	
Navigation_perso_CI		tx	tx	x	
Textbaustein	Textbausteine	tx	tx	x	
[+] Rechte					
Neues Template anlegen				Template hochladen	

Abb. 4.6: Dialog Templates bearbeiten

Der Dialog ist in drei Abschnitte gegliedert:

Dateiliste

Diese Liste enthält die Namen (Spalte ganz links) und die Beschreibungen (zweite Spalte von links) aller bereits eingerichteten Templates. Zusätzlich befinden sich hier Buttons für die Funktionen **Bearbeiten**, **Rechte** und **Löschen**. Der Button **Bearbeiten** öffnet den internen Editor, in dem Templates editiert werden. Mit Hilfe des Buttons **Zugriffsrechte** wird ein Dialog geöffnet, in dem Leserechte und Rechte zum Ändern für einzelne Templates gesetzt werden. Über den Button **Löschen** wird das entsprechende Template im System und im Dateisystem gelöscht.

Rechteverwaltung

In diesem Abschnitt, der beim Laden des Template-Managements geschlossen ist und sich erst nach einem Klick auf das Pluszeichen komplett öffnet, werden Schreib-, Lese- und Löschrechte für mehrere Templates gesetzt. Lesen Sie hierzu Abschnitt [4.11 Dateirechte in Imperia verwalten](#) auf Seite 75.

Links

Unterhalb der Rechteverwaltung befinden sich die Links **Neues Template anlegen** und **Template hochladen**. Lesen Sie zu diesen Funktionen Abschnitt 4.3.1 **Neues Template anlegen** auf Seite 63 und Abschnitt 4.3.2 **Template hochladen** auf Seite 64.

4.3.1 Neues Template anlegen

Um ein Template neu anzulegen, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 4.3 **Templates** auf Seite 62 beschrieben.
 2. Klicken Sie auf den Link **Neues Template anlegen**. Es wird folgender Dialog geöffnet:

► Neues Template anlegen

Name:	Template für Nachrichten
Beschreibung:	Eingabe - Template für Nachrichtenartikel

Abb. 4.7: Dialog Neues Template anlegen

Tragen Sie in die einzelnen Felder die entsprechenden Daten ein. Imperia generiert anhand dieser Daten eine neuen Template und legt dieses im Verzeichnis /site/templates ab.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

Die neue Datei wird in die Dateiliste eingetragen und kann von dort über den Button **Bearbeiten** im internen Editor geöffnet und editiert werden (siehe Abschnitt 4.3.4 **Template bearbeiten** auf Seite 64).

4.3.2 Template hochladen

Templates lassen sich mit einem beliebigen Texteditor erstellen. Führen Sie folgende Schritte aus, um ein extern erstelltes Template auf Ihr Imperia-System hochzuladen:

1. Öffnen Sie das Template-Management wie unter Abschnitt 4.3 **Templates** auf Seite 62 beschrieben.
2. Klicken Sie unterhalb der Rechteverwaltung auf den Link **Template hochladen**.

Es wird folgender Dialog geöffnet:

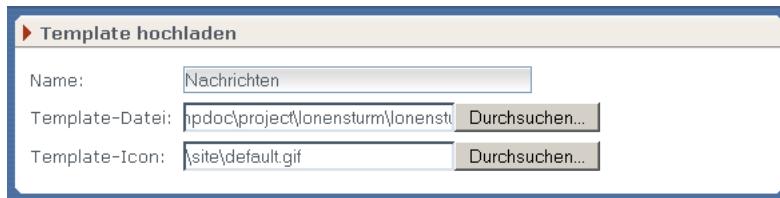


Abb. 4.8: Dialog Template hochladen

3. Tragen Sie in die Felder die entsprechenden Daten ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie ein bestehendes Template im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis /site/templates kopiert und steht anschließend in Imperia zur Verfügung.



Hinweis:

Ähnlich wie Metadateien sollten Templates im UTF-8 Zeichensatz angelegt werden. Bei entsprechenden Vorkehrungen ist dies aber nicht zwingend notwendig. Lesen Sie hierzu auch Abschnitt 3.10.9 Unicode in Imperia auf Seite 41

4.3.3 Template-Vorschau

Um ein Template in einer Vorschau anzeigen zu lassen, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 4.3 **Templates** auf Seite 62 beschrieben.
2. Klicken Sie in der linken Spalte der Dateiliste auf den Namen des Templates.

Es öffnet sich ein weiteres Fenster, in dem das Template angezeigt wird. In dieser Vorschau wird kein Template-Code (IF-Abfragen, Ersetzungen, Perl-Code o.ä.) ausgeführt.

4.3.4 Template bearbeiten

Um ein Template zu bearbeiten, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 4.3 **Templates** auf Seite 62 beschrieben.
2. Betätigen Sie den Button **Bearbeiten** in der entsprechenden Zeile der Dateiliste. Damit rufen Sie den internen Editor auf, in dem Sie den Templatecode verändern können.

4.3.5 Template löschen

Um ein Template zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das Template-Management wie unter Abschnitt 4.3 **Templates** auf Seite 62 beschrieben.
2. Klicken Sie auf den Button **Löschen** neben dem entsprechenden Eintrag in der Dateiliste. Daraufhin erscheint eine Sicherheitsabfrage, die Sie mit **Okay** bestätigen müssen. Anschließend wird das Template aus dem System und damit auch aus dem Verzeichnis `/site/templates` gelöscht. Der Button **Zurück** bricht die Aktion ab.

4.4 Flexmodule

Mit dem Flexmodul-Management verwalten Sie Flexmodule. Über diesen Dialog können Sie Flexmodule bearbeiten, löschen, anlegen und aus dem lokalen Dateisystem in das Imperia-System laden.

Das Flexmodul-Management öffnen Sie über **Hauptmenü -> Struktur -> Flexmodule**. Sie gelangen zu folgendem Dialog:

Flexmodule bearbeiten						
	Typ	Beschreibung	Anzeigen bei ...	Ablehnen bei ...	Bearbeiten	Löschen
attach	htms	Anhang	de	en	¶	×
attach_en	htms	Attachment	en	de	¶	×
bild_fliess	htms	Bild mit Fließtext	de	en	¶	×
bild_links	htms	Bild Text	de	en	¶	×
bild_links_en	htms	Picture Text	en	de	¶	×
bild_rechts	htms	Text Bild	de	en	¶	×
bild_rechts_en	htms	Text Picture	en	de	¶	×
celum_350prop	htms	celum 350xprop	de	en	¶	×
check_form	perl	Form-Checkbox	form	noform	¶	×
input_form_kurz	htms	Form-Inputfeld (kurz)	form	noform	¶	×
input_form_lang	htms	Form-Inputfeld (lang)	form	noform	¶	×

Neues Flex-Modul erzeugen Flex-Modul hochladen

Abb. 4.9: Dialog Flexmodule bearbeiten

Der Dialog zeigt in einer Liste alle im System vorhandenen Flexmodule. Zu jedem Flexmodul werden in Spalten folgende Informationen angezeigt:

Name

Diese Spalte enthält den Dateinamen des Flexmoduls.

Typ

Abhängig davon, ob das Flexmodul in HTML- oder Perl-Code geschrieben ist, kann in dieser Spalte entweder `htms` oder `perl` stehen.

Beschreibung

Hier wird die Beschreibung angezeigt, die das Feld `DESCRIPTION` im Quellcode des Flexmoduls enthält.

Anzeigen bei ...

Diese Spalte enthält etwaige VALIDCODES des Flexmoduls. Lesen Sie hierzu auch den Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ablehnen bei ...

Hier werden etwaige INVALIDCODES des Flexmoduls angezeigt. Lesen Sie hierzu Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ferner stehen Buttons und Links zur Verfügung, mit deren Hilfe Flexmodule editiert oder gelöscht sowie neu erstellt und in das System hochgeladen werden können.

4.4.1 Neues Flexmodul erzeugen

Um ein neues Flexmodul zu erzeugen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Flexmodul-Management** wie unter Abschnitt 4.4 **Flexmodule** auf Seite 65 beschrieben.
2. Klicken Sie auf den Link **Neues Flexmodul erzeugen**. Es öffnet sich folgender Dialog:

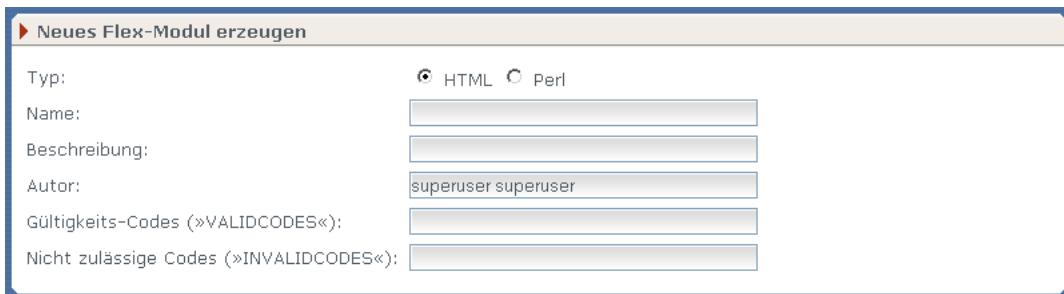


Abb. 4.10: Dialog Neues Flexmodul erzeugen

3. Tragen Sie die entsprechenden Informationen in die einzelnen Felder ein. Die Felder "Gültigkeits-Codes" und "Nicht zulässige Codes" nehmen VALID- bzw. INVALIDCODES auf, mit denen Sie die Verfügbarkeit von Flexmodulen in Templates steuern. Für weitere Einzelheiten lesen Sie den Abschnitt **Zugriff auf Flexmodule einschränken** im Kapitel **Flexmodule** des Programmierhandbuchs.
4. Klicken Sie auf den Button **Okay**, um Ihre Eingaben zu speichern. Der Button **Schließen** schließt den Dialog, ohne Eingaben zu speichern.

4.4.2 Flexmodule hochladen

Flexmodule können Sie auch außerhalb von Imperia mit einem beliebigen Texteditor anlegen und in das System hochladen. Gehen Sie dazu wie folgt vor:

1. Rufen Sie das **Flexmodul-Management** auf wie unter Abschnitt 4.4 **Flexmodule** auf Seite 65 beschrieben.
2. Klicken Sie auf den Link **Flexmodul hochladen**. Folgender Dialog öffnet sich:

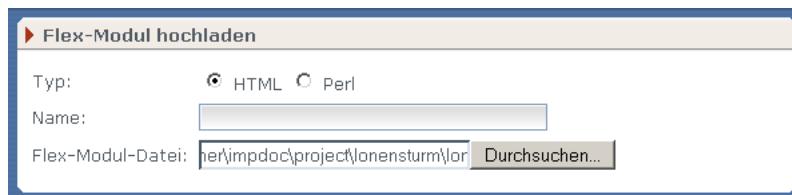


Abb. 4.11: Dialog Flexmodul hochladen

3. Tragen Sie die entsprechenden Informationen in die Felder ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie ein bestehendes Flexmodul im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis /site/flex kopiert und steht anschließend in Imperia zur Verfügung.



Hinweis:

Wenn Sie ein extern erzeugtes Flexmodul im Verzeichnis site/flex speichern, ohne es mit Hilfe des hier beschriebenen Dialogs hochzuladen, erscheint es ebenfalls automatisch in der Liste der Flexmodule.



Hinweis:

Beachten Sie, dass Flexmodule im UTF-8 Zeichensatz (ohne BOM) erstellt sein müssen.

4.4.3 Flexmodule bearbeiten

Flexmodule können Sie mit jedem Text-Editor bearbeiten. Sie liegen im Verzeichnis `/site/flex` und haben je nach Inhalt die Dateiendung `.htms` (Flexmodule mit HTML-Code) oder `.perl`, bzw. `.pl` (Flexmodule mit Perl-Code).

Zum direkten Editieren in Imperia steht der interne Editor zur Verfügung. Um ein Flexmodul mit dem internen Editor zu editieren, gehen Sie wie folgt vor:

1. Öffnen Sie das **Flexmodul-Management** wie unter Abschnitt 4.4 **Flexmodule** auf Seite 65 beschrieben.
 2. Klicken Sie auf den Button **Bearbeiten** rechts neben dem entsprechenden Flexmodul.
- Es wird der interne Editor gestartet. Zur Bedienung des internen Editors lesen Sie bitte Kapitel **Der interne Editor** im Userhandbuch.
3. Ändern Sie das Flexmodul und speichern Sie die Änderungen.

4.4.4 Flexmodule löschen

Um ein Flexmodul zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Flexmodul-Management** wie unter Abschnitt 4.4 **Flexmodule** auf Seite 65 beschrieben.
2. Klicken Sie auf den Button **Löschen** rechts neben dem Flexmodul, das gelöscht werden soll.

Das Flexmodul wird aus der Liste und aus dem Verzeichnis `/site/flex` entfernt.

4.5 Slotmodule

Mit dem Slotmodul-Management verwalten Sie Slotmodule. Über diesen Dialog können Sie Slotmodule bearbeiten, löschen, anlegen und aus dem lokalen Dateisystem in das Imperia-System laden.



Hinweis:

*Einzelheiten zur Syntax und zur Handhabung von Slotmodulen finden Sie im Programmierhandbuch im Kapitel **Flexmodule und Slots**.*

Das Slotmodul-Management öffnen Sie über **Hauptmenü -> Struktur -> Slotmodule**. Sie gelangen zu folgendem Dialog:

	Typ	Beschreibung	Anzeigen bei ...	Ablehnen bei ...	Bearbeiten	Löschen
drittel	htms	33-33-33	termine	✓	X	
dump	perl	Dump	termine	✓	X	
fifti	htms	50-50	news,termine	✓	X	
fifti_not_preindexed	htms	50-50 ohne Indexvorbelegung	termine	✓	X	

Abb. 4.12: Dialog **Slotmodule bearbeiten**

Der Dialog zeigt in einer Liste alle im System vorhandenen Slotmodule. Zu jedem Slotmodul werden in Spalten folgende Informationen angezeigt:

Name

Diese Spalte enthält den Dateinamen des Slotmoduls.

Typ

Abhängig davon, ob das Slotmodul in HTML- oder Perl-Code geschrieben ist, steht in dieser Spalte entweder `htms` oder `perl`.

Beschreibung

Hier sehen Sie die Beschreibung, die das Feld `DESCRIPTION` im Quellcode des Slotmoduls enthält.

Anzeigen bei ...

Diese Spalte enthält etwaige VALIDCODES des Slotmoduls. Lesen Sie hierzu auch den Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ablehnen bei ...

Hier sehen Sie etwaige INVALIDCODES des Slotmoduls. Lesen Sie hierzu auch den Abschnitt **Zugriff auf Slots und Flexmodule einschränken** im Kapitel **Flexmodule und Slots** des Programmierhandbuchs.

Ferner stehen Buttons und Links zum Editieren, Löschen, neu Erstellen und Hochladen von Slotmodulen zur Verfügung.

4.5.1 Neues Slotmodul erzeugen

Um ein neues Slotmodul zu erzeugen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Slotmodul-Management** wie unter Abschnitt 4.5 **Slotmodule** auf Seite 67 beschrieben.
2. Klicken Sie auf den Link **Neues Slotmodul erzeugen**. Es öffnet sich folgender Dialog:

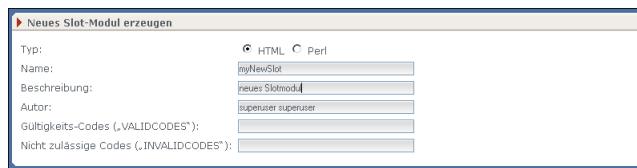


Abb. 4.13: Dialog Neues Slotmodul erzeugen

3. Tragen Sie die entsprechenden Informationen in die einzelnen Felder ein. Die Felder "Gültigkeits-Codes" und "Nicht zulässige Codes" nehmen VALID- bzw. INVALIDCODES auf, mit denen Sie die Verfügbarkeit von Slotmodulen in Templates steuern. Für weitere Einzelheiten lesen Sie den Abschnitt **Zugriff auf Slotmodule einschränken** im Kapitel **Slotmodule** des Programmierhandbuchs.
4. Klicken Sie auf den Button **Okay**, um Ihre Eingaben zu speichern. Der Button **Schließen** schließt den Dialog, ohne Eingaben zu speichern.

4.5.2 Slotmodule hochladen

Slotmodule können Sie auch außerhalb von Imperia mit einem beliebigen Texteditor anlegen und in das System hochladen. Gehen Sie dazu wie folgt vor:

1. Rufen Sie das **Slotmodul-Management** auf wie unter Abschnitt 4.5 **Slotmodule** auf Seite 67 beschrieben.
2. Klicken Sie auf den Link **Slotmodul hochladen**. Folgender Dialog öffnet sich:

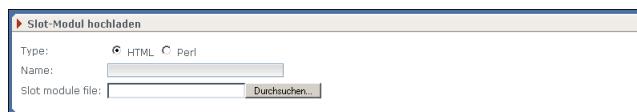


Abb. 4.14: Dialog Slotmodul hochladen

3. Tragen Sie die entsprechenden Informationen in die Felder ein. Mit Hilfe des Buttons **Durchsuchen...** können Sie ein bestehendes Slotmodul im Dateisystem lokalisieren und den Pfad in das entsprechende Feld des Dialogs eintragen.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs. Die eingetragene Datei wird in das Verzeichnis /site/Slot kopiert und steht anschließend in Imperia zur Verfügung.



Hinweis:

Wenn Sie ein extern erzeugtes Slotmodul im Verzeichnis site/Slot speichern, ohne es mit Hilfe des hier beschriebenen Dialogs hochzuladen, erscheint es ebenfalls automatisch in der Liste der Slotmodule.



Hinweis:

Beachten Sie, dass Slotmodule im UTF-8 Zeichensatz (ohne BOM) erstellt sein müssen.

4.5.3 Slotmodule bearbeiten

Slotmodule können Sie mit jedem Text-Editor bearbeiten. Sie liegen im Verzeichnis /site/Slot und haben je nach Inhalt die Dateiendung .htms (Slotmodule mit HTML-Code) oder .perl, bzw. .pl (Slotmodule mit Perl-Code).

Zum direkten Editieren in Imperia steht der interne Editor zur Verfügung. Um ein Slotmodul mit dem internen Editor zu editieren, gehen Sie wie folgt vor:

1. Öffnen Sie das **Slotmodul-Management** wie unter Abschnitt 4.5 **Slotmodule** auf Seite 67 beschrieben.
2. Klicken Sie auf den Button **Bearbeiten** rechts neben dem entsprechenden Slotmodul.

Es wird der interne Editor gestartet. Zur Bedienung des internen Editors lesen Sie bitte Kapitel **Der interne Editor** im Userhandbuch.

3. Ändern Sie das Slotmodul und speichern Sie die Änderungen.

4.5.4 Slotmodule löschen

Um ein Slotmodul zu löschen, gehen Sie wie folgt vor:

1. Öffnen Sie das **Slotmodul-Management** wie unter Abschnitt 4.5 **Slotmodule** auf Seite 67 beschrieben.
2. Klicken Sie auf den Button **Löschen** rechts neben dem Slotmodul, das gelöscht werden soll.

Das Slotmodul wird aus der Liste und aus dem Verzeichnis /site/Slot entfernt.

4.6 Codeinclude-Dateien

In separaten Dateien gespeicherte Code-Bausteine lassen sich über die Imperia-Benutzeroberfläche anlegen, editieren und verwalten. Sie können außerdem Code-Bausteine hochladen, die Sie außerhalb von Imperia erstellt haben.

Mehr über die Erstellung von Codeincludes lesen Sie im Abschnitt "Codeincludes" im Programmierhandbuch.

Das Codeinclude-Management erreichen Sie, wenn Sie den Menüpunkt **Hauptmenü -> Struktur -> Codeinclude-Dateien** auswählen.

▶ Codeincludes bearbeiten		
Ort: Include	Bearbeiten	Löschen
<input type="checkbox"/> perso	▼	X
<input type="checkbox"/> formcheck.htms	▼	X
<input type="checkbox"/> stylesheets.htms	▼	X
<input type="checkbox"/> forminfo.pl	▼	X
<input type="checkbox"/> html_edit.pl	▼	X
<input type="checkbox"/> Metainfo.pl	▼	X
<input type="checkbox"/> Alle auswählen	X Auswahl löschen	
Neues Codeinclude oder Verzeichnis		Codeinclude-Datei hochladen

Abb. 4.15: Der Dialog Codeincludes bearbeiten

Sie erhalten eine Übersicht des Inhalts des Verzeichnisses site/include auf Ihrem Produktionssystem. Dabei werden die Dateinamen vollständig inklusive der Dateiendung angezeigt.

Zur besseren Übersichtlichkeit können Sie Ihre Includes in Unterverzeichnissen strukturieren. Verzeichnisnamen sind in der Liste fett dargestellt und verlinkt. Klicken Sie einen Verzeichnisnamen an, um dessen Inhalt als Dateiliste anzeigen zu lassen.

Wenn Sie sich in einem Unterverzeichnis befinden, ist der erste Eintrag in der Dateiliste immer ein Link auf die nächsthöhere Verzeichnisebene. Dieser Link ist durch zwei Punkte kenntlich gemacht.

Alle Funktionen, die Sie im Dialog "Codeincludes bearbeiten" auswählen, werden in dem Verzeichnis ausgeführt, dessen Inhalt gerade in der Übersicht angezeigt wird.

4.6.1 Neues Codeinclude / Verzeichnis erstellen

Um ein neues Codeinclude oder Verzeichnis anzulegen, führen Sie folgende Schritte aus:

1. Öffnen Sie das Codeinclude-Management wie unter Abschnitt 4.6 **Codeinclude-Dateien** auf Seite 69 beschrieben.
2. Klicken Sie auf den Link **Neues Codeinclude oder Verzeichnis**. Sie gelangen zu folgendem Dialog:



Abb. 4.16: Der Dialog Neue Codeinclude-Datei anlegen

3. Wählen Sie über die Radiobuttons aus, ob Sie ein *HTML*- oder *Perl*-Codeinclude bzw. ein *Verzeichnis* erstellen möchten.



Hinweis:

Imperia ergänzt automatisch die jeweiligen Dateiendungen für HTML- und Perl-Includes. Sie können diese also bei der Eingabe des Dateinamens weglassen.

4. Geben Sie auch einen kurzen *Beschreibungstext* und einen *Namen* ein.
5. Bestätigen Sie Ihre Eingaben durch einen Klick auf **Okay** oder verwerfen Sie sie durch einen Klick auf **Schließen**. Wenn Sie Ihre Eingaben verwerfen, werden Sie automatisch zum Dialog **Codeincludes bearbeiten** zurückgeführt.

Sollte bereits ein Verzeichnis oder Include mit dem von Ihnen eingegebenen Namen bestehen, erscheint eine Warnmeldung, wenn Sie Ihre Eingaben mit *Okay* bestätigen.

Geben Sie dann einen anderen Namen ein oder aktivieren Sie die Option zum Überschreiben bestehender Dateien, indem Sie die Checkbox zwischen den beiden Eingabefeldern markieren.



Hinweis:

Verzeichnisse können auf diese Weise nicht überschrieben werden, um den Verlust von Include-Dateien durch das Überschreiben von Verzeichnissen, die Includes enthalten, zu verhindern.

Nachdem Sie Ihre Eingaben bestätigt haben, werden Sie automatisch zum Dialog **Codeincludes bearbeiten** zurückgeführt. Ihre neu erstellte Datei bzw. Ihr neu erstelltes Verzeichnis erscheint nun in der Übersicht.



Hinweis:

*Eine mit Imperia neu erstellte Codeinclude-Datei ist lediglich ein Grundgerüst ohne Inhalt. Sie müssen diese Dateien noch editieren (siehe Abschnitt 4.6.3 **Codeinclude-Dateien bearbeiten** auf Seite 72).*

4.6.2 Codeinclude-Dateien hochladen

Grundsätzlich können Sie Codeincludes mit einem beliebigen Texteditor erstellen. Außerhalb von Imperia angelegte Codeinclude-Dateien müssen Sie in Imperia hochladen, um Sie im System verfügbar zu machen. Führen Sie dazu folgende Schritte aus:

1. Öffnen Sie das Codeinclude-Management wie unter Abschnitt 4.6 **Codeinclude-Dateien** auf Seite 69 beschrieben.
2. Klicken Sie auf **Codeinclude-Datei hochladen**. Dieser Link führt Sie zu folgendem Dialog:



Abb. 4.17: Der Dialog Codeinclude-Datei hochladen

3. Bestimmen Sie den Typ der hochzuladenden Datei durch Klick auf einen der Radiobuttons. Zur Verfügung stehen die Typen *HTML* und *Perl*. Entsprechend Ihrer Auswahl hängt Imperia automatisch die passende Dateiendung an den Dateinamen an.
4. Tragen Sie in das Eingabefeld *Name* einen Dateinamen für Ihre Include-Datei ein. Ohne Angabe eines Dateinamens wird der Upload-Vorgang mit einer Warnmeldung abgebrochen.
Mit dem Button *Durchsuchen* öffnen Sie ein Dialogfenster zur Auswahl der hochzuladenden Codeinclude-Datei.
5. Bestätigen Sie Ihre Eingaben durch Klick auf den Button **Okay** oder verwerfen Sie sie, indem Sie auf **Schließen** klicken. Sie werden dann automatisch zum Dialog **Codeincludes bearbeiten** zurückgeführt.

Sollte bereits eine Codeinclude-Datei mit dem von Ihnen eingegebenen Namen bestehen, wird eine Warnmeldung angezeigt und der Upload nicht ausgeführt. Wie bei der Erstellung von Includes oder Verzeichnissen, können Sie auch hier eine Option zum Überschreiben bestehender Dateien aktivieren.

Nach dem Hochladen der Datei erscheint wieder der Dialog **Codeincludes bearbeiten** mit aktualisierter Dateiliste.

4.6.3 Codeinclude-Dateien bearbeiten

Codeincludes lassen sich folgendermaßen über die Imperia-Benutzeroberfläche bearbeiten:

1. Rufen Sie zunächst das Codeinclude-Management auf (siehe Abschnitt 4.6 **Codeinclude-Dateien** auf Seite 69).
2. Klicken Sie auf den Bearbeiten-Button  neben dem Dateinamen des Includes, das Sie bearbeiten wollen. Daraufhin öffnet sich ein Popup-Fenster mit dem internen Imperia-Editor, in dem Sie die ausgewählte Datei editieren können. Für eine genaue Beschreibung des internen Editors lesen Sie bitte den Abschnitt **Der interne Editor** im Benutzerhandbuch.

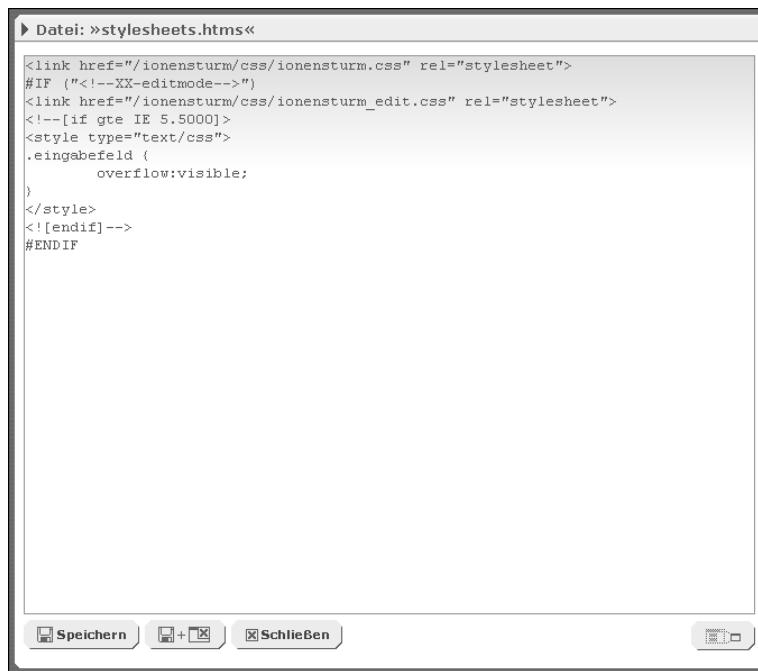


Abb. 4.18: Das Bearbeitungsfenster für Codeincludes

Sie können sowohl extern als auch mit Imperia erstellte (siehe Abschnitt 4.6.1 **Neues Codeinclude / Verzeichnis erstellen** auf Seite 70) Codeincludes bearbeiten.

4.6.4 Codeinclude-Dateien / Verzeichnisse löschen

Führen Sie die folgenden Schritte aus, um ein Codeinclude oder ein Verzeichnis für Codeinclude-Dateien über die Imperia-Benutzeroberfläche zu löschen:

1. Öffnen Sie das Codeinclude-Management wie unter Abschnitt 4.6 **Codeinclude-Dateien** auf Seite 69 beschrieben.
2. Klicken Sie auf den Löschen-Button  neben einem Codeinclude oder Verzeichnis, um dieses zu löschen.
3. Wenn Sie mehrere Includes bzw. Verzeichnisse entfernen wollen, markieren Sie zunächst die Checkboxen links daneben und klicken Sie anschließend auf den Button Auswahl löschen .

4.7 Dynamische Module

Dynamische Module, die in der Datei `dynamic.conf` konfiguriert werden, werden ausführlich in dem Kapitel **Dynamische Module** des Programmierhandbuchs behandelt. Um die Datei `dynamic.conf` zu editieren, steht der interne Editor zur Verfügung. Die Datei `dynamic.conf` liegt im Verzeichnis `/site/config`. Gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Dynamische Module** im Ordner **Struktur**.
Es wird der interne Editor geöffnet und die vorhandene `dynamic.conf` wird angezeigt.
2. Ändern Sie ggfs. den Code im Textfeld.
3. Speichern Sie die Änderungen über den Button **Okay** unterhalb des Dialogs.

4.7.1 Laden einer externen Datei in den Editor

Um eine außerhalb von Imperia erzeugte Datei in den Editor zu laden, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Dynamische Module** im Ordner **Struktur**.
2. Geben Sie den Pfad und Dateinamen der gewünschten Datei in das Feld **Datei hochladen** ein oder benutzen Sie den Button **Durchsuchen...**, um einen Dialog zum Durchsuchen Ihrer Festplatten zu öffnen.
3. Speichern Sie die Datei über den Button **Okay** am unteren Rand der Imperia-Oberfläche.

Die externe Datei wird jetzt als `dynamic.conf` gespeichert und die bereits vorhandene Datei wird überschrieben.

4.8 Template-Reparser

Der Template-Reparser bietet die Möglichkeit, geänderte Templates erneut zu parsen, so dass alle Dokumente, die mit einer früheren Version dieses Templates erzeugt wurden, erneut erzeugt werden.

Hierbei können sowohl Dokumente, die sich bereits auf einem Zielsystem befinden, als auch solche, die in der Freischaltliste angezeigt werden, neu erstellt werden.

Zusätzlich wird bestimmt, ob die neu erstellten Dokumente nach dem Reparsen in einen Workflow einfließen, der Freischaltliste hinzugefügt oder automatisch auf ein Zielsystem freigeschaltet werden sollen.

Um den Template-Reparser zu verwenden, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Template-Reparser** im Ordner **Struktur**.
Es wird folgender Dialog in einem neuen Fenster geöffnet:



Abb. 4.19: Dialog Template-Reparser

Der Dialog des Template-Reparsers gliedert sich in drei Abschnitte:

Template zum Reparsen auswählen

Hier werden die Templates ausgewählt, die neu geparst werden sollen.

Dokumente zum Reparsen auswählen

In diesem Abschnitt wird ausgewählt, ob die Dokumente vom Zielsystem und / oder die aus der Freischaltliste neu erstellt werden sollen.

Aktion nach dem Reparsen

In diesem Abschnitt wird bestimmt, wie mit den neu erstellten Dokumenten verfahren werden soll. Es kann zwischen den Aktionen **Dokumente in den Workflow zurückführen**, **Dokumente der Freischaltliste hinzufügen** und **Dokumente automatisch freischalten** gewählt werden.

2. Markieren Sie im Abschnitt **Template zum Reparsen auswählen** die Templates, die neu geparst werden sollen. Mehrere Templates werden durch zusätzliches Drücken der Strg-Taste markiert.

Der Link **Alle auswählen** markiert alle Templates in der Liste.



Achtung:

Wenn Sie alle Templates markieren, dauert das Reparen, wenn viele Dokumente neu erstellt werden müssen, unter Umständen sehr lange. Infolgedessen kann es zu Server-Timeouts und somit zum vorzeitigen Abbruch des Vorganges kommen.

Für das Reparen größerer Dokumentenmengen steht ein Kommandozeilen-Tool zur Verfügung, das in Abschnitt 3.16 Reparen umfangreicher Sites auf Seite 52 beschrieben ist.

3. Wählen Sie im Abschnitt **Dokumente zum Reparsen auswählen**, welche Dokumente neu erstellt werden sollen.
4. Wählen Sie im Abschnitt **Aktion nach dem Reparsen** die Aktion aus, die nach dem Reparen ausgeführt werden soll.
5. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

Das Reparen wird gestartet und die Dokumente entsprechend Ihrer Auswahl weiterverarbeitet.

4.9 Assets verschieben

Diese Funktion ist ausführlich in Abschnitt 10.1.4 **Einzelne Assets und Asset-Rubriken verschieben** auf Seite 134 beschrieben.

4.10 Systemdateien abgleichen

Mit dieser Funktion können Sie geänderte Systemdateien auf verschiedenen Zielsystemen übertragen. Dies ist dann notwendig, wenn beispielsweise Systemdienste, die den Hintergrund-Daemon auf einem Zielsystem betreffen (SiteActive), geändert wurden.

Gehen Sie wie folgt vor:

1. Legen Sie eine Datei `/site/config/system_files.conf` an und tragen Sie dort die zu kopierenden Dateien inklusive des Pfades ausgehend vom SITE-Verzeichnis ein. Im Standard wird dort die Datei `/config/imperia.schedule` angegeben, in der die Systemdienste abgespeichert werden.
2. Klicken Sie im Ordner **Struktur** auf den Menüpunkt **Systemdateien abgleichen**.
Es öffnet sich der Dialog **Systemdateien kopieren** in einem neuen Fenster.
3. Markieren Sie in der Liste **Vorhandene Zielsysteme** die Zielsysteme, auf die die geänderten Systemdateien übertragen werden sollen.
4. Klicken Sie auf den Button **Okay**. Der Abgleich wird durchgeführt und es wird ein Protokoll des Vorgangs angezeigt. Der Button **Schließen** schließt den Dialog, ohne dass ein Abgleich durchgeführt wird.

4.11 Dateirechte in Imperia verwalten

Imperia bietet die Möglichkeit, einzeln Rechte pro Template-, Meta- und Workflow-Datei zu vergeben. Hierzu werden einer Rolle Schreib-, Lösch- oder Leserechte für jede Datei einzeln zugewiesen oder entzogen. Diese Funktion kann gleichzeitig für mehrere Rollen und mehrere Dateien eines Typs durchgeführt werden. Im Folgenden beschreiben wir die nötigen Schritte am Beispiel der Metadateien. Für alle anderen Dateitypen sind die Schritte analog durchzuführen.

Um die Rechte für Imperia-Dateien zu vergeben, gehen Sie wie folgt vor:

1. Öffnen Sie den Management-Dialog für die zu editierenden Dateien.
2. Klicken Sie auf das Pluszeichen mit der Beschriftung **Rechte** unterhalb der Dateiliste.

Folgender Dialog öffnet sich:

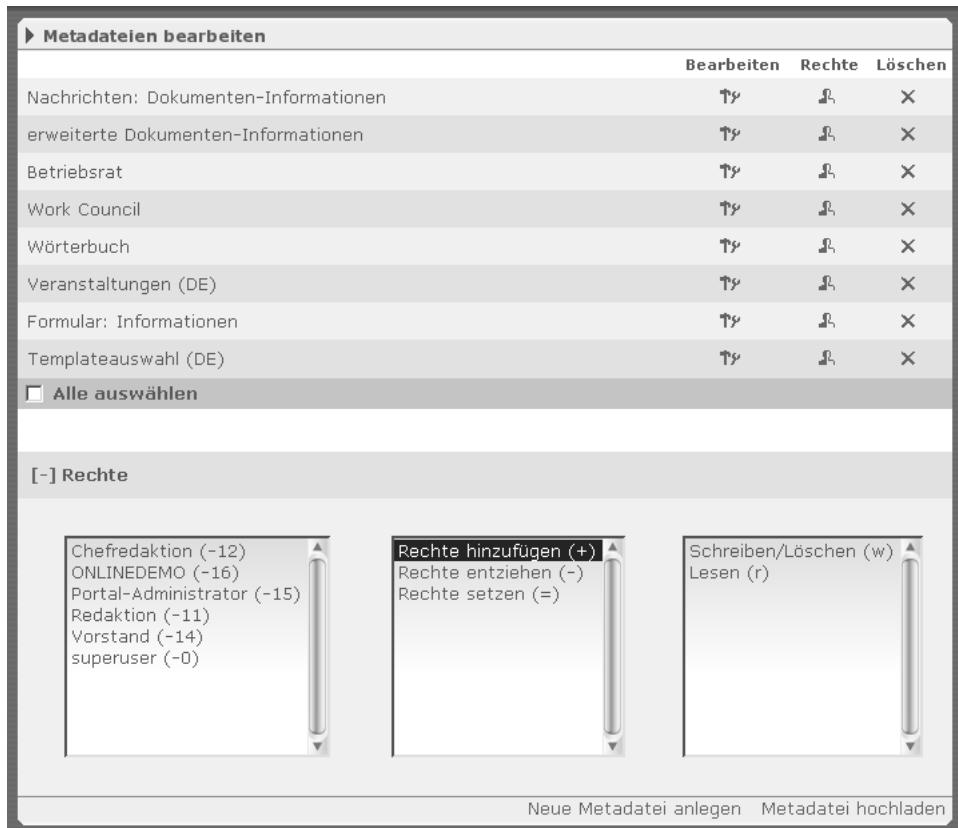


Abb. 4.20: Dialog Rechtevergabe, Beispiel Metadateien

Der Dialog gliedert sich in folgende Abschnitte:

Dateiliste

Die Dateiliste wird in dieser Ansicht um Checkboxen ergänzt. Einzelne Dateien werden mit Hilfe der jeweiligen Checkbox markiert, alle Dateien können mit einem Klick über die Checkbox "Alle auswählen" markiert werden.

Rechte

In diesem Abschnitt werden die Rechte für die Rollen vergeben. Der Abschnitt ist in drei Felder unterteilt. Das linke Feld enthält die Liste aller im System eingerichteten Rollen, auf die man als angemeldeter User Zugriff hat. Das mittlere Feld enthält alle verfügbaren Funktionen und das rechte Feld alle Rechte, die vergeben werden können.

3. Markieren Sie in der Dateiliste mit Hilfe der Checkboxen die Dateien, für die Rechte gesetzt werden sollen.
4. Markieren Sie im Abschnitt **Rechte** in der linken Liste die Rollen, deren Rechte geändert bzw. gesetzt werden sollen. Mehrere Einträge werden mit Hilfe der Strg-Taste markiert.
5. Wählen Sie im Abschnitt **Rechte** in der mittleren Liste die Funktion aus, die Sie ausführen wollen. Die Funktionen bedeuten im Einzelnen:

Rechte hinzufügen (+)

Diese Funktion fügt die ausgewählten Rechte den bereits bestehenden Rechten der ausgewählten Rollen hinzu.

Rechte entziehen (-)

Hiermit werden die markierten Rechte den markierten Rollen entzogen.

Rechte setzen (=)

Diese Funktion setzt die Rechte der ausgewählten Rollen auf die markierten Rechte. Nicht markierte Rechte werden den Rollen entzogen.

6. Markieren Sie in der rechten Liste die Rechte, die geändert bzw. eingestellt werden sollen.
7. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

Kapitel 5. IWE - Imperia-WYSIWYG-Editor

Der IWE (Imperia-WYSIWYG-Editor) ist ein WYSIWYG Editor auf Basis des FCK-Editors der durch entsprechende Anpassungen auf die Anforderungen des Imperia CMS angepasst wurde.

Aktuelle Informationen zum FCK finden Sie unter www.fckeditor.net (Stand der Quellenangabe: Juli 2009).

5.1 Migration vom EWE-Modul zum IWE

Wenn Sie bisher den Easy Web Editor (EWE) als WYSIWYG-Editor verwendet haben, können Sie den damit erzeugten Inhalt durch den IWE bearbeiten. Eine Änderung im Template-Code ist hierfür nicht notwendig. Der IWE verfügt über einen Kompatibilitätsmodus, der die EWE-Template-Syntax einschließlich vieler Aufrufparameter unterstützt. Die Kompatibilität kann je nach Implementierung variieren. Fügen Sie zur Aktivierung des EWE-Kompatibilitätsmodus folgende Variable in der Datei `site/config/system.conf` hinzu:

```
"EWE_COMPAT_MODE" = "1"
```

Durch den „EWE_COMPAT_MODE“ werden EWE-Templateaufrufe als IWE-Aufrufe interpretiert.

Außerdem müssen Sie das EWE-Modul deaktivieren. Kommentieren Sie dazu in der Datei `site/config/system.conf` folgenden Eintrag aus:

```
"TEMPLATE_EWE" = "yes"
```

Der Imperia Template-Prozessor stellt nun anstelle des EWE den IWE dar.

5.2 Konfiguration des IWE

Der IWE bietet eine Vielzahl von Konfigurationsoptionen, mit welchen die Funktionalitäten, das Aussehen und das Verhalten beeinflusst werden.

Es gibt es zwei Möglichkeiten den Editor zu konfigurieren:

- Konfiguration im Template

Bei dieser Methode erfolgt die Konfiguration der jeweiligen Instanz des Editors über das Imperia-Template.

Die Verwendung des IWE im Template können Sie im, Kapitel **IWE - Imperia-WYSIWYG-Editor** im Programmierhandbuch nachschlagen.

- Steuerung über eine Konfigurationsdatei

Hierbei wird entweder für jede Instanz eine eigene Konfigurationsdatei im Imperia-Template spezifiziert, oder es wird eine globale Datei für alle Instanzen verwendet.



Wichtig:

Bitte beachten Sie die Hierarchie der Konfigurationsmöglichkeiten. Wenn Sie eine Konfigurationsoption im Template gesetzt haben, hat diese grundsätzlich Vorrang vor der gesetzten Option in der Konfigurationsdatei. Es gibt also insgesamt drei Hierachiestufen:

1. *Template*
2. *Benutzerdefinierte Konfigurationsdatei der Instanz*
3. *Globale Konfigurationsdatei*

Die globale Konfigurationsdatei finden im Dateisystem unter:

5.3 Konfigurationsoptionen

Der IWE bietet weitere Konfigurationsoptionen. Es wird unterschieden zwischen:

- Template-Only Optionen

Dabei handelt es sich um Konfigurationsoptionen, welche nur im Template gesetzt werden können. Diese Optionen sind speziell für die Verwendung des Editors in Imperia.

- Freie Optionen

Dies sind Optionen (FCK-Optionen), welche sowohl im Template als auch in den Konfigurationsdateien verwendet werden können.

5.3.1 Template-Only Optionen

Template-Only Optionen haben die Besonderheit, dass sie, wie der Name bereits vermuten lässt, nur in der Processing Instruction im Template gesetzt werden können. Folgende Optionen gibt es:

- Width

Mit der Option „Width“ definieren Sie die Breite des Editorfensters.

Mögliche Werte sind numerische Werte in Pixel. Beispiel:

```
<?imperia iwe
    id: iwe1
    Width: 570
?>
```

- Height

Mit der Option „Height“ definieren Sie die Höhe des Editorfensters.

Mögliche Werte sind numerische Werte in Pixel. Beispiel:

```
<?imperia iwe
    id: iwe1
    Height: 400
?>
```

- External

Die Option „External“ schaltet den External-Mode an bzw. ab. Ein aktiver External-Mode führt dazu, dass der Editor in einem eigenen Popup-Fenster geöffnet wird und nicht eingebettet im Template erscheint.

Mögliche Werte sind: '1' (an) oder '0' (aus). Beispiel:

```
<?imperia iwe
    id: iwe1
    External: 1
?>
```

- ExternalStyle

Die Option „ExternalStyle“ definiert das Aussehen des Platzhalters im Template, welcher den Inhalt anzeigt, der über den Editor im External-Mode gepflegt wird. Als möglicher Wert kann CSS-Code verwendet werden. Ebenso ist die Angabe einer CSS-Datei möglich, welche die Klasse 'imperia_external_editor' beinhaltet. Beispiel:

```
<?imperia iwe
    id: iwe1
    External: 1
    ExternalStyle: background: red
?>
```

- Content

Die Option 'Content' bietet die Möglichkeit, die jeweilige Instanz des IWE vorab mit Inhalt zu befüllen.

Ein möglicher Wert ist jede Art von Text (String) oder eine Imperia-Variable. Beispiel:

```
<?imperia iwe
    id: iwe1
    Content: Das ist ein Beispiel
?>
```

5.3.2 Freie Optionen

Der IWE bietet eine große Menge an Möglichkeiten, die Funktionalität an eigene Bedürfnisse anzupassen. Im folgenden Abschnitt werden einige der wichtigsten Optionen genannt, jedoch nicht alle. Eine komplette Übersicht über alle verfügbaren Konfigurationsoptionen finden Sie im Internet unter:

http://docs.cksource.com/FCKeditor_2.x/Developers_Guide/Configuration/Configuration_Options

Im folgenden Abschnitt wird nur die Syntax für die Verwendung in einer Konfigurationsdatei verwendet.

5.3.2.1 Verhalten des Editors

- BaseHref

Die Option 'BaseHref' funktioniert wie der HTML-Tag '<base href="">'. Sie definiert die Basis-URL aller verwendeten Links und Bilder.

Als möglicher Wert kann eine URL gesetzt werden. Beispiel:

```
FCKConfig.BaseHref = 'http://www.mysite.com/';
```

- DocType

Die Option 'DocType' ermöglicht die Angabe des DOCTYPE, welches in dem editierbaren Teil verwendet wird. Er spezifiziert wie gerendert wird.

Als gültigen Wert ist eine DOCTYPE-Syntax anzugeben. Das folgende Beispiel lässt den Editor den eingegebenen Inhalt als 'XHTML 1.0 Transitional' rendern:

```
FCKConfig.DocType = '<!DOCTYPE html PUBLIC
"-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">;
```

- MaxUndoLevels

Die Option 'MaxUndoLevels' setzt die maximale Anzahl von 'Undo' bzw. 'Rückgängig' Schritten, die der User nutzen kann. Der Standardwert ist 15.

Beispiel:

```
FCKConfig.MaxUndoLevels = 20;
```

- StartupFocus

Wenn die Option 'StartupFocus' auf 'true' gesetzt wird, erzwingt der Editor, dass der Eingabecursor innerhalb des Eingabebereichs erscheint. Dies ist sinnvoll, wenn sich nur eine Instanz des IWE im Template befindet und der User direkt mit dem Tippen beginnt.

Als mögliche Werte stehen 'true' oder 'false' zur Auswahl. Beispiel:

```
FCKConfig.StartupFocus = true;
```

5.3.2.2 Styles

- BodyId und BodyClass

Dieses Optionspaar ermöglicht die Angabe einer ID und/oder einer Klasse, welche zum BODY des zu editierenden Bereichs hinzugefügt wird.

Beispiel: Ihre Hintergrundfarbe (background-color) ist schwarz (#000), doch der Inhalt, welcher über den Editor gepflegt wird, befindet sich in einem '<div id="news">' Element und nutzt dieses CSS:

```
body { background-color:#000; } h1 { color:#FFF; }
#news { background-color:#FFF; } #news h1 { color:#333; }
```

Wenn Sie also in Ihrer Konfigurationsdatei die folgende Zeile verwenden:

```
FCKConfig.BodyId = 'news' ;
```

wird der Bereich, welcher durch den Editor gepflegt wird, einen weissen Hintergrund haben. Ebenfalls werden alle H1-Elemente in dunkler Farbe dargestellt. Diese Option bietet also die Möglichkeit, die gleiche CSS-Datei für die gesamte Seite und für den Editor zu verwenden.

- CustomStyles

Die Option 'CustomStyles' bietet Ihnen die Möglichkeit, eigene Styles in das 'Styles'-Menue im Editor hinzuzufügen. Das folgende Beispiel zeigt, wie ein neuer Style angelegt wird.

Beispiel: Sie wollen einen neuen Style hinzufügen, welcher 'MY STYLE 1' heisst und folgende Eigenschaften hat:

- Der Style beinhaltet das Format Überschrift 2 (h2)
- Die Schriftfarbe ist blau (Blue)
- Die Hintergrundfarbe ist rot (Red)

Um den benutzerdefinierten Style 'MY STYLE 1' hinzuzufügen, benötigen Sie den folgenden Code in Ihrer Konfigurationsdatei:

```
FCKConfig.CustomStyles = { 'MY STYLE 1' : {Element
:'h2', Styles : {'color' : 'Blue' , 'background-color' : 'Red' } } };
```

Wie Sie sehen, ist die Syntax der Option recht einfach:

- Innerhalb eines Apostrophenpaars (') wird der Name des Styles gesetzt.

- Nach dem Doppelpunkt wird die Definition des Styles in geschweifte Klammern ({ }) geschrieben.
- 'Element' zeigt auf das Element, für welches der Style übernommen werden soll (z.B. h1, h2, h3).
- 'Styles' definiert, welche Eigenschaften Ihr benutzerdefinierter Style haben soll (color, background-color, font-family).

• FontColors

Die Option 'FontColors' definiert die Farben, die im 'Text Color' und im 'Background Color' Menue aufgelistet werden.

Als gültige Werte können HTML-Farbcodes im HEX Format angegeben werden (FFFFFF). Beispiel:

```
FCKConfig.FontColors =
'000000,993300,333300,003300,003366,000080,333399,333333,800000,
FF6600,808000,808080,008080,0000FF,666699,808080,FF0000,FF9900,
99CC00,339966,33CCCC,3366FF,800080,999999,FF00FF,FFCC00,FFFF00,
00FF00,00FFFF,00CCFF,993366,C0C0C0,FF99CC,FFCC99,FFFF99,CCFFCC,
CCFFFF,99CCFF,CC99FF,FFFFFF';
```

• FontFormat

Die Option 'FontFormat' definiert die Schriftformate (z.B. Überschrift 1, Überschrift 2), welche im 'Format' Menue aufgelistet sind.

Als gültige Werte können HTML-Formate angegeben werden (z.B. p, h1, div). Beispiel:

```
FCKConfig.FontFormats ='p;h1;h2;h3;h4;h5;h6;pre;address;div';
```

• FontNames

Die Option 'FontFormat' definiert die Schriftarten, welche im 'Font' Menue aufgelistet sind.

Beispiel:

```
FCKConfig.FontNames= 'Arial;Comic Sans MS;Courier New;
Tahoma;Times New Roman;Verdana';
```

• FontSizes

Die Option 'FontSizes' definiert die Schriftgrößen, welche im 'Size' Menue aufgelistet sind. Standardmäßig sind folgende Werte verfügbar:

```
FCKConfig.FontSizes =
'smaller;larger;xx-small;x-small;small;medium;large;x-large;xx-large';
```

Es kann jeder gültige CSS-kompatible Wert genutzt werden. Das folgende Beispiel setzt Pixel, 'em' und Prozent-Größen:

```
FCKConfig.FontSizes = '14px;1.5em;80%';
```

Für jede Grösse kann auch ein Name vergeben werden, der anstatt des Wertes im 'Size' Menue angezeigt wird. Name und Wert müssen durch einen Schrägstrich getrennt werden. Beispiel:

```
FCKConfig.FontSizes = '14px/Fourteen
Pixels;1.5em/A bit bigger;80%/A bit smaller' ;
```

5.3.2.3 HTML Ausgabe

- FillEmptyBlocks

Dieser Parameter füllt leere Tabellenzellen mit "Whitespaces" auf, um einen Zellenumbruch zu verhindern. Er ist standardmäßig gesetzt.

```
FCKConfig.FillEmptyBlocks = false  
;
```

5.3.2.4 Benutzer-Interface

- AutoDetectLanguage

Die Option 'AutoDetectLanguage' gibt dem Editor die Möglichkeit, die Systemsprache zu erkennen. Wenn die Option auf 'true' gesetzt ist, wird der Editor in der selben Sprache, wie die des Browsers oder des Betriebssystems, angezeigt.

Als gültiger Wert kann 'true' oder 'false' gesetzt werden.

```
FCKConfig.AutoDetectLanguage = true;
```

- DefaultLanguage

Die Option 'DefaultLanguage' spezifiziert die Sprache, welche vom IWE als Standard verwendet wird.

Gültige Werte sind die jeweiligen Landeskennungen (z.B.: Deutsch (de), Französisch (fr), Italienisch (it), usw.)

```
FCKConfig.DefaultLanguage = "de";
```

- Plugging.add

Mit der Option 'Plugin.add' wird ein neuer Eintrag in das Kontextmenü des Editors eingefügt.

Im folgendem Beispiel werden Einträge zum Einfügen von Tabellen in das Kontextmenü eingebettet.

```
FCKConfig.Plugins.Add( 'tbrow', null, sPluginPath );
```

- ToolbarSets

Mit dieser Option ist es möglich, die Symbolleiste (Toolbar) entsprechend anzupassen.

Nach einer initialen Installation des IWE sieht die Toolbar wie folgt aus:

```
FCKConfig.ToolbarSets["Imperia"] = [  
    ['Source', '-', 'Preview', '-', 'Templates'],  
    ['Cut', 'Copy', 'Paste', 'PasteText', 'PasteWord', '-', 'Print', 'SpellCheck'],  
    ['Undo', 'Redo', '-', 'Find', 'Replace', '-', 'SelectAll', 'RemoveFormat'],  
    ['Form', 'Checkbox', 'Radio', 'TextField', 'Textarea', 'Select', 'Button',  
     'MAMAssetButton', 'HiddenField'],  
    '/',  
    ['Bold', 'Italic', 'Underline', 'StrikeThrough', '-', 'Subscript', 'Superscript'],  
    ['OrderedList', 'UnorderedList', '-', 'Outdent', 'Indent', 'Blockquote', 'CreateDiv'],  
    ['JustifyLeft', 'JustifyCenter', 'JustifyRight', 'JustifyFull'],  
    ['MAMAssetLink', 'Unlink', 'Anchor'],  
    ['MAMAsset', 'Table', 'Rule', 'Smiley', 'SpecialChar', 'PageBreak'],  
    '/', ['Style', 'FontFormat', 'FontName', 'FontSize'],  
    ['TextColor', 'BColor'], ['FitWindow', 'ShowBlocks']]
```

Mit dieser Option besteht zum Beispiel die Möglichkeit, Einträge zum Einfügen von Tabellen hinzufügen, wie im folgendem Beispiel aufgeführt.

```
FCKConfig.ToolbarSets['TableTest'] = [  
    'tbrow', 'tbcoll', 'tbtab'  
];
```

Kapitel 6. Rubriken

Mit Hilfe von Rubriken wird ein Projekt in verschiedene Themenbereiche unterteilt. Die am häufigsten verwendete Unterteilungsstrategie ist die Differenzierung nach Themen- oder Sachgebieten. Jedem Themenbereich wird eine eigene Rubrik zugeordnet, in der wiederum Unterrubriken für einzelne Themen erstellt werden. Unterrubriken können beliebig verschachtelt werden.

Beispielsweise kann eine Rubrik *Sport*, eine Rubrik *Reisen* und eine Rubrik *News* erstellt werden. Als Unterrubriken für die Hauptrubrik *Sport* könnten zum Beispiel die Rubriken *Segeln*, *Golf* und *Fußball* existieren. Die Unterrubrik *Fußball* könnte weiter unterteilt werden in *Bundesliga* und *Champions League*.

Rubriken können entweder nur weitere Unterrubriken oder Unterrubriken und Dokumente enthalten. Rubriken, die nur Unterrubriken enthalten können, werden zur Strukturierung des Rubrikenbaums verwendet. Wird diese Eigenschaft von **enthält Unterrubriken und Dokumente** zu **enthält Unterrubriken** geändert, kann dies zu folgendem Problem führen:

- Es können keine Dokumente mehr aus dem Archiv oder vom Zielsystem in diese Rubrik importiert werden. Der importierende User erhält eine Fehlermeldung (Keine Berechtigung, um Dokumente in dieser Rubrik zu erzeugen!).

Da im Allgemeinen nicht jeder User Dokumente in jeder Rubrik erstellen oder bearbeiten soll, werden beim Erzeugen einer Rubrik die Zugriffsrechte bestimmt. Das heißt, es wird festgelegt, welche Rolle einem User Zugriffsrechte auf die Rubrik verleiht.

Wählt ein User den Menüpunkt **Neues Dokument erstellen** aus, sieht er im folgenden Dialog nur die Rubriken, in denen er aufgrund seiner aktuellen Rolle Dokumente erstellen darf. Wichtig hierbei ist, dass er auf alle Oberrubriken dieser Rubriken Leserechte haben muss.

Zugriffsrechte können in vier Abstufungen erteilt werden:

- Lesen (Inhalt der Rubrik anzeigen lassen),
- Dokumente erzeugen,
- Dokumente löschen und
- Rubrik bearbeiten.

Weiterhin werden folgende Eigenschaften eines Dokuments durch die Rubrik bestimmt:

- die standardmäßig verwendete Metadatei,
- das standardmäßig verwendete Template und
- der verwendete Workflow.

6.1 Rubriken-Eigenschaften bestimmen

Die Eigenschaften einer Rubrik stellen Sie im Dialog **Rubriken-Eigenschaften** ein. Um diesen Dialog zu öffnen wählen Sie **Hauptmenü -> Struktur -> Rubriken**. Sie gelangen zur **Rubriken-Verwaltung**. Dort können Sie eine neue Rubrik erstellen oder eine bestehende Rubrik bearbeiten.

Der Dialog ist in folgende Abschnitte unterteilt:

- Rubriken-Information
- Workflow-Auswahl
- Metadateien und Templates auswählen
- Zugriffsrechte
- Asset-Rubriken

Rubriken-Eigenschaften können in einem Template verwendet werden. Lesen Sie hierzu Kapitel **Eigenschaften der Rubrik auslesen** im Programmierhandbuch.

6.1.1 Rubriken-Informationen angeben

Die Rubriken-Informationen dienen als Default-Einstellung für die in einer Rubrik erzeugten Dokumente. Sie können durch Workflow-Plug-Ins jederzeit während des Workflows verändert werden, so dass ein Dokument beispielsweise nicht unbedingt in dem hier angegebenen Verzeichnis gespeichert wird.

Folgende Informationen geben Sie in diesem Abschnitt des Dialogs ein:

- Tragen Sie im Feld **Name** den Namen der Rubrik ein.

Dieser Name erscheint in der Rubrikenliste und für jedes Dokument in der entsprechenden Spalte der Schreibtisch-Ansicht.

- Tragen Sie im Feld **Beschreibung** eine kurze Beschreibung der Rubrik ein.

Die Beschreibung erscheint im Dialog **Rubriken-Eigenschaften** und wird im Archiv verwendet.

- Geben Sie das Verzeichnis an, in dem alle in dieser Rubrik erzeugten Dokumente gespeichert werden sollen.

Dieser Wert kann, während das Dokument den Workflow durchläuft, geändert werden. Es ist nicht erlaubt, Leerzeichen in Verzeichnisnamen zu verwenden.

- Tragen Sie im Feld **Dateiname** den Default-Dateinamen für Dokumente ein, die in dieser Rubrik erzeugt werden.

Dieser Wert kann, während das Dokument den Workflow durchläuft, geändert werden. Es ist nicht erlaubt, Leerzeichen in Dateinamen zu verwenden.

- Meta-Information: Klicken Sie auf den Link Bearbeiten, um zusätzliche Meta-Informationen für die Rubrik einzugeben. Diese sind dann für alle Dokumente dieser Rubrik sowie für anschließend erzeugt Unterrubriken gültig. Bereits bestehende Unterrubriken "erben" nachträglich hinzugefügte Meta-Informationen nicht! Lesen Sie hierzu auch Abschnitt 4.1.2 **Rubriken bearbeiten** auf Seite 56.

Hinweis:



Diese Option steht nur bei der nachträglichen Bearbeitung bestehender Rubriken zur Verfügung.

- Soll die neue Rubrik nur zur Strukturierung des Rubrikenbaums dienen, wählen Sie die Option **nur Unterrubriken**. In einer solchen Rubrik können nur Unterrubriken, jedoch keine Dokumente angelegt werden.

Achtung:



*Wird die Option **nur Unterrubriken** ausgewählt, können keine Dokumente, die ursprünglich in dieser Rubrik erzeugt wurden, zurückimportiert werden. Dies greift auch dann, wenn Dokumente aus dem Archiv reimportiert werden sollen.*

Die Option **Dokumente und Unterrubriken** erlaubt sowohl die Erzeugung von Dokumenten in dieser Rubrik als auch das Anlegen von Unterrubriken.

6.1.2 Workflow festlegen

Der Workflow, der für die Rubrik verwendet werden soll, wird im Abschnitt **Workflow festlegen** des Dialogs **Rubriken-Eigenschaften** bestimmt. Da ein Workflow mehrere Einstiegspunkte haben kann, wird nicht nur der Workflow, sondern auch der Einstiegspunkt bestimmt.

Jedes in der Rubrik erzeugte Dokument erhält eine Kopie des hier eingestellten Workflows und durchläuft alle zu diesem Workflow gehörenden Schritte, bis es den Workflow verlässt. Dokumente, die sich innerhalb des Workflows befinden, werden von Änderungen dieses Workflows nicht beeinflusst.

In der Pull-down-Liste werden alle Einstiegspunkte aller zur Verfügung stehenden Workflows in der Form

Name des Workflows -> Name des Einstiegspunkts

aufgelistet.

Wählen Sie den gewünschten Workflow-Einstiegspunkt aus.

Ein Dokument, das in der Rubrik erstellt wird, durchläuft alle Workflow-Schritte, die dem ausgewählten Workflow-Einstiegspunkt folgen.

6.1.3 Metadatei und Template bestimmen

In diesem Abschnitt werden die Metadatei und das Template bestimmt, die in der Rubrik verwendet werden sollen.

Diese Einstellungen können über eine Template-Auswahl während der Eingabe der Metainformationen jederzeit überschrieben werden. In der Pull-down-Liste **Metadatei** werden alle Metadateien aufgelistet, die sich im Verzeichnis /site/metafiles befinden.

Wählen Sie die Metadatei aus, die in der Rubrik verwendet werden soll.



Hinweis:

Sollte im ausgewählten Workflow kein MetaEdit-Schritt verwendet werden, muss trotzdem eine Metadatei ausgewählt werden.

In der Pull-down-Liste **Template** sind alle Templates aufgelistet, die sich im Verzeichnis /site/templates befinden. Der Eintrag in der Liste wird aus der Templatebeschreibung und der Templatenummer (in Klammern) gebildet.

Wählen Sie das Template aus, das in der Rubrik verwendet werden soll.



Hinweis:

Sowohl Metadatei als auch Template können im Laufe des Workflows gewechselt werden.

Lesen Sie hierzu auch Abschnitt 8.5.6 Bearbeiten auf Seite 112, Abschnitt 8.5.17 Metaedit auf Seite 120, sowie Abschnitt 8.5.18 Metasetter auf Seite 121.

6.1.4 Rubriken-Zugriff

Im Folgenden wird beschrieben, wie der Zugriff auf Rubriken eingeschränkt werden kann. Das Zugriffsrecht auf eine Rubrik kann in vier verschiedenen Abstufungen vergeben werden:

- Lesen (des Rubrikinhaltes),
- Dokumente erzeugen,
- Dokumente löschen und
- Rubrik bearbeiten.

Für jede dieser Stufen können Rollen bestimmt werden, die Zugriff auf die Rubrik erhalten.

Das Recht, in einer Rubrik Dokumente erstellen zu können, beinhaltet gleichzeitig das Recht, Dokumente, die in dieser Rubrik erstellt wurden, freizuschalten. Das Recht, den Menüpunkt **Dokumente freischalten** im Hauptmenü sehen zu können, muss ebenfalls für die betreffende Rolle über die Menü-Verwaltung gesetzt sein.

6.1.4.1 Zugriffsrechte zuweisen

Zugriffsrechte für Rubriken werden im Abschnitt **Rechte bearbeiten** des Dialogs **Rubriken-Eigenschaften** eingestellt. Um einer Rolle Zugriffsrechte mit einer bestimmten Stufe zu erteilen, gehen Sie wie folgt vor:

1. Markieren Sie in dem Auswahlfeld **Verfügbare Rollen** einen oder mehrere Einträge.
Mehrere Einträge werden mit Hilfe der Strg-Taste markiert.

2. Klicken Sie auf den zwischen den Auswahlfeldern stehenden nach rechts weisenden Pfeil-Button.

Die ausgewählten Rollen werden in die Liste **Gewählte Rollen** übertragen.

Nimmt ein User eine der ausgewählten Rollen an, so erhält er auf die Rubrik mit der entsprechenden Stufe Zugriff.

6.1.4.2 Zugriffsrecht aberkennen

Um ein zugewiesenes Zugriffsrecht wieder abzuerkennen, gehen Sie wie folgt vor:

1. Markieren Sie in dem Auswahlfeld **Gewählte Rollen** die Rollen, die nicht mehr zum Zugriff mit der entsprechenden Stufe berechtigen sollen.

Mehrere Einträge werden mit Hilfe der Strg-Taste markiert.

2. Klicken Sie auf den zwischen den Auswahlfeldern stehenden nach links weisenden Pfeil-Button.

Die markierten Rollen werden aus der Liste **Gewählte Rollen** entfernt.

6.1.5 Asset-Rubriken

Eine Asset-Rubrik besitzt alle Eigenschaften einer herkömmlichen Dokumenten-Rubrik in Imperia. Sie können in den Einstellungen einer Asset-Rubrik Metainformationen hinterlegen, die für alle darin angelegten Assets verfügbar sind. Die Details zu den Asset-Rubriken finden Sie in Abschnitt [10.1 Mit Asset-Rubriken arbeiten](#) auf Seite 132.

6.1.5.1 Asset-Rubriken in 'Freigeschaltete Dokumente bearbeiten' anzeigen

Im **Freigeschaltete Dokumente bearbeiten** Dialog gibt es die Möglichkeit, die Liste der angezeigten Dokumente auf bestimmte Rubriken zu beschränken. Damit in der Auswahlliste auch die gewünschte Asset-Rubrik angezeigt wird, muss folgende Metavariable in der entsprechenden Rubrik hinzugefügt werden:

- `__imperia_show_in_cat_filter = „1“`

Kapitel 7. User und Rollen

Die User-Verwaltung ermöglicht es, abhängig vom Umfang der erworbenen Lizenz User und Rollen anzulegen und zu verwalten.

Bevor User und Rollen angelegt werden empfehlen wir, ein Konzept über die zukünftige Struktur des Systems zu erstellen. Je genauer die Struktur geplant wird, desto einfacher und unkomplizierter ist nicht nur die erste Konfiguration, sondern auch die spätere Erweiterung und Pflege. Folgende Überlegungen sollten in die Struktur einfließen:

Wie ist der Arbeitsablauf innerhalb des Betriebes organisiert?

Die Organisation und die Arbeitsweise Ihres Unternehmens kann durch die User- und Rollen-Struktur in Imperia abgebildet werden. Die hier gefundenen Zusammenhänge geben erste Hinweise auf Art und Anzahl der Rollen.

Welcher Mitarbeiter nimmt welche Aufgaben wahr?

Anhand dieser Überlegungen können für die einzelnen Mitarbeiter die Rollen bestimmt werden, die ihm zur Verfügung stehen.

Welche Rollen müssen aufgrund der beiden vorherigen Überlegungen angelegt werden?

Die hier definierten Rollen werden später in der User-Verwaltung angelegt (siehe Abschnitt 7.1 **Neue Rolle anlegen** auf Seite 89).

Welche Inhalte sollen von welchen Rollen bearbeitet werden?

Diese Überlegungen fließen bei der Vergabe der Zugriffsrechte auf die Rubriken und damit auf die in dieser Rubrik erstellten Dokumente ein. Es ist auf jeden Fall sinnvoll, Zugriffsrechte auf Rubriken restriktiv zu handhaben, um ungewollte bzw. unerlaubte Änderungen zu verhindern. Die hier definierten Rubriken-Rollen-Beziehungen werden über den Rubriken-Zugriff (siehe Abschnitt 6.1.4 **Rubriken-Zugriff** auf Seite 87) umgesetzt.

Zugriffsrechte auf bestimmte Bereiche oder Funktionen können nur Rollen zugewiesen werden. Ein User hat, mit Ausnahme des Superusers, keine Rechte, sondern erhält alle Zugriffsrechte über die aktuelle Rolle, die er gerade angenommen hat.

Rollen dienen unter anderem dazu, die gleichen Zugriffsrechte mehreren Usern zugänglich zu machen. Einem User können auch mehrere Rollen zugewiesen werden, die er nach Belieben wechseln kann. Er hat jedoch immer nur die Zugriffsrechte, die ihm durch seine aktuell angenommene Rolle verliehen werden.

Rollen können beliebig verschachtelt werden. In der daraus resultierenden Struktur vererbt eine Rolle, die weitere Rollen enthält, ihre Zugriffsrechte auf ihre Unterrollen.

Wir empfehlen folgendes Vorgehen:

Erstellen Sie zuerst die gewünschte Rollenstruktur und geben Sie den einzelnen Rollen die entsprechenden Rechte für die Menüs, die Rubriken und die Objekte in der Mediendatenbank. Lesen Sie hierzu folgende Abschnitte:

- Abschnitt 9.2.3 **Zugriffsrechte für Menüs einstellen** auf Seite 130,
- Abschnitt 6.1.4 **Rubriken-Zugriff** auf Seite 87 und
- Abschnitt 10.4.17 **Zugriffsrechte für Asset-Rubriken verwalten** auf Seite 145.

Jeder User kann über den Link "Rollen" in der Fußzeile jederzeit eine andere Rolle annehmen und so seine aktuellen Zugriffsrechte ändern. Lesen Sie hierzu Kapitel **Rolle annehmen** im Userhandbuch.

7.1 Neue Rolle anlegen

Neue Rollen werden im Dialog **Rollen-Einstellungen** angelegt. Dieser Dialog wird über den Menüpunkt **Neue Rolle anlegen** im Ordner **User und Rollen** geöffnet.

Der Dialog ist in mehrere Abschnitte gegliedert:

Obligatorische Angaben

Hier wird der Name der neuen Rolle eingetragen. Der Name ist ein Pflichtfeld, ohne den die neue Rolle nicht gespeichert werden kann.

Optionale Angaben

Hier können im Feld **Kommentar** optionale Angaben wie Bemerkungen o.ä. gemacht werden.

Rollenzugehörigkeit

In diesem Abschnitt werden die Rollenzugehörigkeiten der neuen Rolle eingestellt. Wenn Sie eine neue Rolle einer anderen Rolle zuordnen, wird die neue Rolle eine Unterrolle der ausgewählten Rolle und erbt deren Zugriffsrechte. Alle User, die die neue Rolle annehmen können, erben diese Rechte ebenfalls. Die gewünschten Rollen werden in der linken Liste markiert und durch einen Klick auf den nach rechts weisenden Pfeil-Button in die rechte Liste übertragen. Der nach links weisende Pfeil-Button entfernt markierte Einträge aus der rechten Liste wieder.

Zugriffsrechte

Dieser Abschnitt wird nur angezeigt, wenn Sie bereits neue Rollen angelegt haben. Wählen Sie hier die Rollen aus, die das Änderungsrecht für die neue Rolle bekommen sollen. Die gewünschten Rollen werden in der linken Liste markiert und durch einen Klick auf den nach rechts weisenden Pfeil-Button in die rechte Liste übertragen. Der nach links weisende Pfeil-Button entfernt markierte Rollen aus der rechten Liste. Wird die Checkbox **Alle Rollen dürfen auf diese Rolle zugreifen** markiert, dürfen alle Rollen die neue Rolle ändern.

Globale Rechte für ACL-gesteuerte Objekte

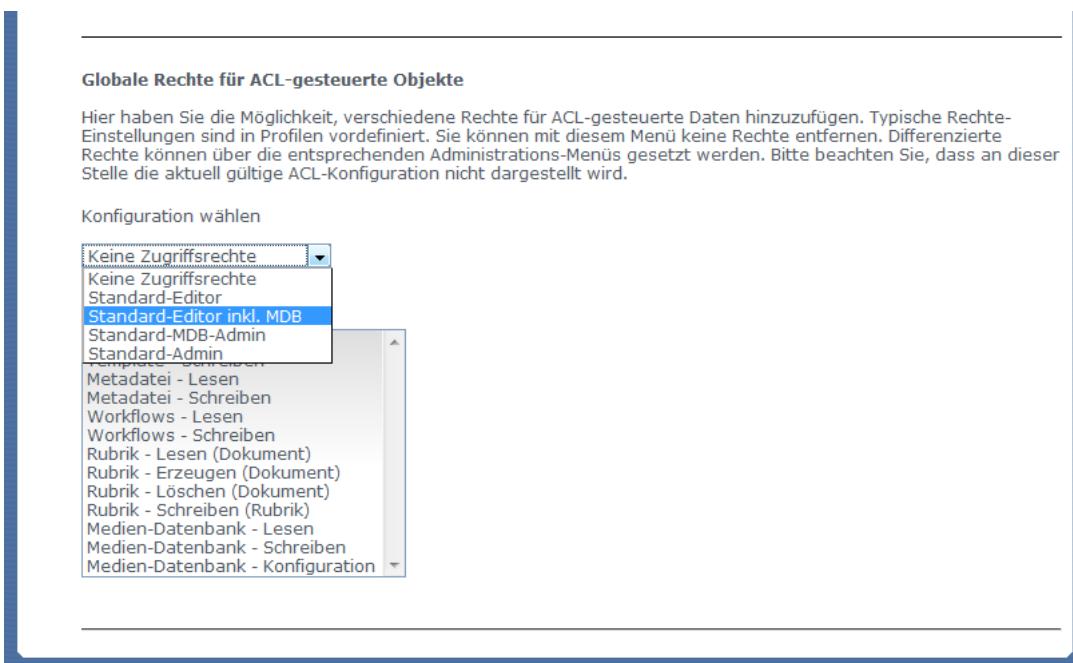


Abb. 7.1: Dialog Globale Rechte für ACL-gesteuerte Objekte

Imperia verwaltet einige Zugriffsberechtigungen intern zentral über so genannte Access Control Lists (kurz ACL). In diesem Abschnitt legen Sie die Berechtigung der betreffenden Rolle auf diese Objekte fest. Es wird jedoch nicht die aktuell eingestellte ACL-Konfiguration dargestellt. In dem Drop-Down-Menü **Konfiguration wählen** können Sie einige vordefinierte Berechtigungsprofile auswählen. Bei der Auswahl eines Profils werden die entsprechenden Rechte in der Liste **Selektierte Zugriffsrechte** angezeigt. Diese Profile können nicht verändert werden, sondern dienen nur als Vorlagen. Eine Ergänzung ist jedoch möglich. Markieren Sie dazu weitere Rechte mit gedrückter Strg-Taste.



Hinweis:

Mit dieser Einstellung legen Sie die Zugriffsrechte auf die angegebenen Objekte **global** für die betreffende Rolle fest.

Die vergebenen Rechte sind immer global und sollten daher sparsam gesetzt werden. Beispiele:

- Template - Lesen: diese Rolle darf alle Templates lesen
- Template - Schreiben: diese Rolle darf alle Templates bearbeiten
- Rubrik - Erzeugen: diese Rolle darf Unterrubriken in allen Rubriken erstellen
- Workflows - Schreiben : diese Rolle darf alle Workflows ändern und löschen

Eingaben werden über den Button **Okay** am unteren Rand des Dialogs gespeichert. Anschließend wird ein Hinweis angezeigt, dass die Daten gespeichert werden.

Der Button **Schließen** schließt den Dialog, ohne dass Änderungen gespeichert werden.

Das Anlegen einer neuen Rolle ist damit abgeschlossen. Um weitere Zugriffsrechte einzurichten, die der neuen Rolle verliehen werden sollen, lesen Sie folgende Kapitel:

- Abschnitt 9.2.3 **Zugriffsrechte für Menüs einstellen** auf Seite 130,
- Abschnitt 6.1.4 **Rubriken-Zugriff** auf Seite 87 und
- Abschnitt 10.4.17 **Zugriffsrechte für Asset-Rubriken verwalten** auf Seite 145.

7.2 Rollendaten bearbeiten

Das Bearbeiten einer Rolle geschieht im Dialog **Rolleneinstellungen**, der über den Menüpunkt **Rollendaten bearbeiten** im Ordner **User und Rollen** geöffnet wird.

Wird dieser Menüpunkt geöffnet, erscheint zunächst eine Liste aller bereits eingerichteten Rollen. Die zu bearbeitende Rolle wird durch einen Klick auf den Namenseintrag in dieser Liste ausgewählt.

The screenshot shows the 'Rollenauswahl' dialog box. It has two main sections:

- Liste einschränken:** This section contains a dropdown menu labeled "Auswahlkriterium" with "all" selected, and a text input field "Beginnt mit" containing "V".
- Rollenliste:** This section displays a table of roles with columns "Name" and "ID". The roles listed are:

Name	ID
Chefredaktion	-12
ONLINEDEMO	-16
Portal-Administrator	-15
Redaktion	-11
Vorstand	-14

Abb. 7.2: Dialog Rollenauswahl

Der Dialog besteht aus zwei Abschnitten:

Liste einschränken

In diesem Abschnitt wird die Anzahl der im Abschnitt **Rolle** angezeigten Rollen begrenzt. Über die Drop-Down-Liste **Beginnt mit** werden die Anfangsbuchstaben der Rollennamen, die angezeigt werden sollen, eingestellt.

Rollen

In diesem Abschnitt werden nur die Rollen angezeigt, die Sie aufgrund Ihrer aktuellen Rolle sehen dürfen und deren Name mit dem im Abschnitt **Liste einschränken** eingestellten Kriterium übereinstimmen.

Die Namen der angezeigten Rollen sind verlinkt, so dass ein Klick auf den Namen den eigentlichen Dialog zum Bearbeiten der Rollendaten öffnet. Neben dem Namen der Rolle zeigt dieser Abschnitt auch die Rollen-ID.

Der Dialog zum Bearbeiten der Rollendaten ist identisch mit dem Dialog zum Anlegen einer neuen Rolle. Ein Beschreibung der einzelnen Abschnitte finden Sie in Kapitel 7.1

Der Button **Okay** am unteren Rand des Dialogs speichert die Änderungen. Es erscheint daraufhin ein Hinweis, dass die Änderungen gespeichert werden. Der Button **Schließen** schließt den Dialog, ohne Änderungen zu speichern.

7.3 Löschen einer Rolle

Wenn Sie eine vorhandene Rolle löschen, verlieren alle Benutzer, die diese Rolle annehmen können, die mit ihr verbundenen Berechtigungen.

Eine Rolle wird über den Button **Löschen** unterhalb des Dialogs **Rolleneigenschaften** gelöscht. Dieser Dialog wird über den Menüpunkt **Rollendaten bearbeiten** im Ordner **User und Rollen** geöffnet.

Wird der Button **Löschen** betätigt, erscheint eine Sicherheitsabfrage, die mit **Okay** bestätigt werden muss. Erst dann wird die ausgewählte Rolle tatsächlich gelöscht.



Achtung:

Wenn Sie nur eine begrenzte Rollenzahl lizenziert haben, sollten Sie keine Rollen löschen, sondern bestehende editieren.

Da aus technischen Gründen einmal vergebene Rollen-IDs beim Löschen der zugehörigen Rolle nicht freigegeben werden, erhöht sich für das System beim Anlegen neuer Rollen auch dann die Gesamtanzahl der Rollen, wenn Sie vorher eine oder mehrere Rollen löschen.

7.4 Neuen User anlegen

Neue User werden im Dialog **Usereinstellungen** angelegt. Dieser Dialog wird über den Menüpunkt **Neuen User anlegen** im Ordner **User und Rollen** geöffnet.

Der Dialog ist in mehrere Abschnitte gegliedert:

Obligatorische Angaben

In diesem Abschnitt werden der Nachname, der Vorname, das Login und das Passwort des neuen Users in die entsprechenden Felder eingetragen. Alle diese Angaben müssen getätigter werden, um den neuen User anlegen zu können. Wird ein Feld nicht ausgefüllt, erscheint beim Versuch zu speichern eine Fehlermeldung. Das gewählte Passwort muss zusätzlich durch eine identische Eingabe in das Feld **Passwort wiederholen** bestätigt werden. Gültige Passwörter umfassen mindestens fünf alphanumerische Zeichen. Passwörter mit weniger Zeichen werden vom System zurückgewiesen. Ferner wird in diesem Abschnitt die Sprache der Imperia-Oberfläche für den neuen User eingestellt. Wird für die Sprache der Eintrag **Standard** ausgewählt, wird die Imperia-Oberfläche in der Standard-Systemssprache angezeigt (siehe Abschnitt 3.10.7 **System-Sprache** auf Seite 41).

Optionale Angaben

In diesem Abschnitt können Adressdaten oder andere Angaben zum neuen User eingegeben werden. Keines der Felder muss in diesem Abschnitt ausgefüllt werden. Diese Felder kann der User über den Menüpunkt **Persönliche Einstellungen** im Ordner **User und Rollen** selber ausfüllen. Über die Programmierung des Templates kann man auf die Werte, die in diesen Feldern gespeichert sind, zugreifen.

Rollenzugehörigkeit

In diesem Abschnitt werden die Rollen ausgewählt, die der neue User annehmen kann. Die Liste **Rollen** enthält die bereits im System angelegten Rollen. Alle Rollen, die der User annehmen darf, müssen von der linken in die rechte Liste übertragen werden. Dies geschieht über den nach rechts weisenden Pfeil-Button, der die in der linken Liste markierten Einträge in die rechte Liste überträgt. Der nach links weisende Pfeil-Button entfernt markierte Einträge in der rechten Liste wieder. Die Reihenfolge der Einträge in der rechten Liste kann über die nach oben bzw. nach unten weisenden Pfeil-Buttons verändert werden.

Zugriffsrechte

Hier wird eingestellt, welchen Rollen das Recht verliehen wird, die Daten des neuen Users zu bearbeiten. Alle Rollen, den dieses Recht verliehen werden soll, müssen von der linken in die rechte Liste übertragen werden. Dies geschieht über den nach rechts weisenden Pfeil-Button, der markierte Einträge von der linken Liste in die rechte Liste überträgt. Der nach links weisende Pfeil-Button entfernt markierte Einträge aus der rechten Liste wieder. Wird die Checkbox **Alle Rollen dürfen auf diesen User zugreifen** markiert, wird allen Rollen das Recht verliehen, die Daten des neuen Users zu bearbeiten.

Die Daten des neuen Users werden mit Hilfe des Buttons **Okay** am unteren Rand des Dialogs gespeichert. Es wird ein Hinweis angezeigt, dass die Userdaten gespeichert werden. Der Button **Schließen** schließt den Dialog, ohne Änderungen zu speichern.

7.5 Userdaten bearbeiten

Das Bearbeiten von Userdaten geschieht über den Dialog **Usereinstellungen**, der über den Menüpunkt **Userdaten bearbeiten** im Ordner **User und Rollen** geöffnet wird.

Wird dieser Menüpunkt geöffnet, erscheint zunächst eine Liste aller bereits im System eingerichteten User. Der zu bearbeitende User wird über einen Klick auf den Namen ausgewählt.

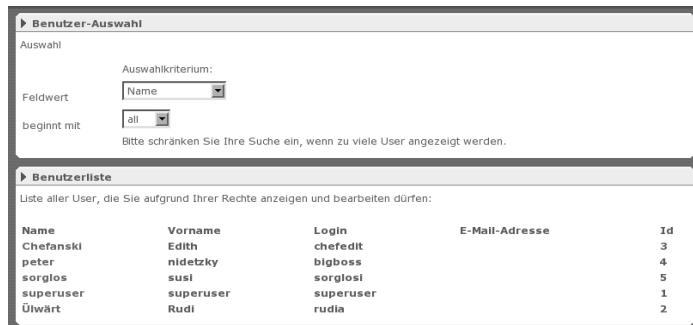


Abb. 7.3: Dialog Userauswahl

Der Dialog besteht aus zwei Abschnitten:

Liste einschränken

In diesem Abschnitt wird die Anzahl der im Abschnitt **User** angezeigten Einträge eingeschränkt. In der Drop-Down-Liste **Feldwert** wird das **Auswahlkriterium** ausgewählt, nach dem die Liste im Abschnitt **User** eingeschränkt werden soll. Mögliche Auswahlen sind *Name*, *Vorname*, *Login* und *Email-Adresse*. Über die Drop-Down-Liste **Beginnt mit** werden die Anfangsbuchstaben der anzuseigenden Einträge ausgewählt. Die Liste im Abschnitt **User** wird automatisch an die Auswahl angepasst. Passt die eingestellte Einschränkung auf keinen Eintrag, bleibt die Liste leer.

User

In diesem Abschnitt werden nur die User angezeigt, die auf die Einschränkung im Abschnitt **Liste einschränken** passen. Ein Klick auf den Namen eines Users öffnet den eigentlichen Dialog zum Bearbeiten der Userdaten, in den die Daten des ausgewählten Users eingetragen werden.

Der Dialog zum Bearbeiten von Userdaten ist identisch mit dem Dialog zum Anlegen eines neuen Users. Die einzelnen Abschnitte des Dialogs werden in Abschnitt 7.4 **Neuen User anlegen** auf Seite 92 beschrieben.

Änderungen an den Userdaten werden über den Button **Okay** am unteren Rand des Dialogs gespeichert. Der Button **Schließen** schließt den Dialog, ohne Änderungen zu übernehmen.

7.6 User löschen

User werden über den Button **Löschen** unterhalb des Dialogs **Usereinstellungen** gelöscht. Dieser Dialog wird über den Menüpunkt **Userdaten bearbeiten** im Ordner **User und Rollen** und über einen anschließenden Klick auf den Namen des entsprechenden Users geöffnet.

Wird der Button **Löschen** betätigt, erscheint eine Sicherheitsabfrage, die mit **Okay** bestätigt werden muss. Erst daraufhin wird der User gelöscht.

7.7 LDAP und Verzeichnisdienste

LDAP steht für Lightweight Directory Access Protocol und ist im Prinzip eine im Netzwerk verteilte Datenbank, die auf dem Client/Server-Prinzip basiert. Ein LDAP-Client öffnet eine Verbindung mit einem LDAP-Server und ruft entweder Informationen ab oder liefert Informationen, die in das Verzeichnis aufgenommen werden sollen. Die Art und Weise, wie dies geschieht, wird durch das Protokoll bestimmt (im Moment Version 3 von LDAP). In diesem Zusammenhang wird LDAP auch als X.500 Lite bezeichnet. X.500 ist ein internationaler Standard für Verzeichnisse, aber wesentlich komplexer aufgebaut als LDAP.

Für die Verwendung von LDAP in Imperia muss das LDAP-Perl-Modul installiert sein. Man kann es unter <http://www.cpan.org/> oder unter <http://perl-ldap.sourceforge.net> herunterladen. In einigen Linux-Distributionen ist es enthalten, muss aber explizit installiert werden.

Da der Begriff *Datenbank* schon gefallen ist und Ähnlichkeiten auch nicht von der Hand zu weisen sind, vergleichen wir im Folgenden die Eigenschaften von LDAP mit den Eigenschaften von Datenbanken und Dateisystemen.

LDAP legt Inhalte optimiert für lesenden Zugriff ab. Datenbanken hingegen bieten beide Varianten: auf Lesezugriff optimiert (z.B.: MySQL) oder auf Schreibzugriff (Oracle). Ferner stellt LDAP keinerlei Transaktionsmechanismen zur Verfügung. Dies ist bei Datenbanken oft ein wesentlicher Sicherheitsaspekt bzgl. Datenverlusten.

Beim Dateisystem liegt der Vergleich deshalb sehr nahe, weil sich Dateisysteme zur Speicherung ihrer Dateien eines Baumes bedienen. Dies ist bei LDAP genauso.

Dateien können im Baum als ein Knoten angesehen werden und haben oft eine Länge von mehreren Gigabyte oder Megabytes. LDAP hingegen ist darauf ausgelegt, dass ein Knoten in seinem Verzeichnisbaum eine eher kleinere Menge an Speicher benötigt. Ein typisches Beispiel für einen Eintrag wären die Daten eines Benutzers, die in der Regel ein Megabyte nicht überschreiten dürfen.

Ein anderer Punkt ist die Art des Zugriffs auf die Daten innerhalb eines Knotens. Bei Dateisystemen sind Zugriffe auf einzelne Bytes der Datei direkt möglich, d.h. ein Byte ist die kleinste lesbare und schreibbare Einheit im Dateisystem. Bei LDAP ist dies nicht möglich. Hier ist die kleinste Einheit, die man schreiben/lesen kann, das Attribut, das später erläutert wird.

Imperia bietet neben der eigenen Rollen- und Userverwaltung die Möglichkeit, an ein LDAP-System angebunden zu werden. Dadurch kann eine bereits vorhandene Rollen- und Userdatenbank in Imperia genutzt werden. Hierbei wird die Administration und Pflege dieses Verzeichnisses nicht mehr von Imperia durchgeführt.

7.7.1 Einschränkungen

Bei der Einbindung eines über LDAP ansprechbaren Verzeichnisdienstes werden bestimmte Funktionen, die im Standard-Paket von Imperia 8 enthalten sind, eingeschränkt:

- Die Administration und Pflege der Rollen- und Userinformationen kann in Imperia nicht mehr in vollem Umfang durchgeführt werden
- Die transitive Rollenstruktur, wie im eigenen Usermanagement von Imperia, wird bei der LDAP-Anbindung nicht aufgelöst. Allerdings ist dieser Punkt bei vielen Verzeichnisdiensten irrelevant, da diese selbstständig ein transitives Rollenmanagement auflösen. Die Rechte bezüglich Zugriff und Modifikation anderer User werden im Rahmen der ersten LDAP-Implementierung nicht verändert. Das heißt im Einzelnen:
 - Ein Leserecht auf alle anderen User und Rollen kann einem einzelnen User nicht aberkannt werden.
 - Bei der Benutzung von ActiveDirectory, dem LDAP-Server von Microsoft®, wird zurzeit nur genau eine Windows®-Domäne unterstützt.

7.7.2 Anbindung an einen Verzeichnisdienst

Zur Anbindung an einen Verzeichnisdienst werden in Imperia zwei Verfahren unterstützt:

Imperia-Account

Dieses Verfahren gewährt dem Imperia-Server einen eigenen Zugang zum Verzeichnisdienst, über den er die User- und Rollendaten auslesen kann. Informationen zu einem User, die Imperia benötigt, aber nicht im Verzeichnisdienst abgelegt werden können, werden im Dateisystem des Imperia-Servers abgelegt.

Benutzer-Account

Bei diesem Verfahren benutzt Imperia den Zugang des jeweils eingeloggten Benutzers, um sich mit dem Verzeichnisdienst zu verbinden. Damit ist gewährleistet, dass jeder Benutzer nur die Informationen sehen kann, die ihm auf Grund seines Accounts dort auch zur Verfügung stehen. Ein Nachteil dieses Verfahrens ist, dass sich Imperia für jeden Benutzer bei jedem Zugriff erneut einloggen muss, wofür Imperia das Passwort des Benutzers im Cookie abspeichern muss. Dieses wird zwar verschlüsselt, es besteht aber trotzdem noch ein gewisses Sicherheitsrisiko.

7.7.3 Verfahren zur Anmeldung an Imperia

Für die Anmeldung an Imperia werden im Rahmen der LDAP-Anbindung mehrere Verfahren unterstützt:

Kerberos

Ist ein Benutzer an einem Kerberos-Server angemeldet, kann er sich mit dem Kerberos-Plug-In durch ein entsprechendes Ticket durch den Server automatisch anmelden lassen. Das Plug-In benötigt auf dem Client ein Programm, das zum Lieferumfang von Imperia gehört. Sobald dieses Programm gestartet wird, ist er ohne Eingabe von Username und Passwort in Imperia unter seiner Identität angemeldet und wird beim Aufruf der Imperia-Anmeldeseite automatisch ins Hauptmenü weitergeleitet. Das Kerberos-Plug-In funktioniert nur mit dem oben beschriebenen Imperia-Account-Verfahren zur Anbindung an den Verzeichnis-Server.

LDAP

Dies ist das originäre Authentifizierungs-Plug-In für die Verzeichnisanbindung an Imperia. Der User gibt sein Passwort für den Verzeichnisdienst bei der Anmeldung an Imperia an. Für die weitere Arbeit mit Imperia kann in der Konfiguration gewählt werden, ob die Anfrage von Daten mit dem User-Account oder einem einheitlichen Imperia-Account gestartet werden.

Passwort

Dies ist das normale Authentifizierungs-Plug-In von Imperia. Hierbei wird für die Anbindung an den Verzeichnisdienst ein Imperia-Account vorausgesetzt. Weiterhin muss in der LDAP-Konfiguration das Passwortfeld angegeben werden. Falls dieses im LDAP-Server kodiert abgelegt wurde, muss in der LDAP-Konfiguration das Schlüsselwort `password-encryption` entsprechend gesetzt werden.

7.7.4 Notwendige Konfiguration für LDAP

Die Konfiguration der LDAP-Anbindung muss zurzeit noch in zwei Dateien manuell vorgenommen werden:

1. In der Datei `/site/config/system.conf`
2. In der Datei `/site/config/ldap.conf`

7.7.4.1 system.conf

In dieser Datei müssen folgende Variablen eingetragen werden:

```
"UMLIGHT_PLUGIN" = "LDAPLight"  
"DATABASE_STORAGE_PLUGIN" = "LDAP"
```

sowie gegebenenfalls noch das verwendete Authentifizierungs-Plug-In (LDAP, Kerberos etc.):

```
"AUTH_PLUGIN" = "LDAP"
```

7.7.4.2 ldap.conf

In dieser Konfigurationsdatei für die LDAP-Anbindung müssen Informationen für den Verzeichnisdienst und Informationen über den LDAP-Zugang, mit dem Imperia auf den Verzeichnisdienst zugreift, gespeichert werden.



Hinweis:

In Abschnitt 13.1 Beispiele für LDAP.conf auf Seite 197 finden Sie LDAP.conf-Beispiele für die Gruppierungsschemata PAM und ADIR.

7.7.4.2.1 Informationen für den Verzeichnisdienst

```
server      ldap.server.net
port        1234
version    3
```

Wird die Variable `version` nicht gesetzt, wird automatisch die Version 2 des LDAP-Protokolls verwendet.

7.7.4.2.2 Informationen über den LDAP-Zugang

Die Informationen über den LDAP-Zugang, mit dem Imperia auf den Verzeichnisdienst zugreift, sind optional für den Fall, dass der *Anonymous Account* des Verzeichnisdienstes keine ausreichenden Leserechte besitzt, um die Imperia-User zu identifizieren. Für spätere Zugriffe außerhalb der Authentifizierungsphase wird die LDAP-Anbindung zwei unterschiedliche Vorgehensweisen unterstützen.

```
bind_dn      cn=Admin, o=Imperia
password    secret
```

- Liegt das Passwort des Imperia-Users vor, dann wird es verwendet, um die User- und Rolleninformationen aus dem Verzeichnisdienst auszulesen.
- Liegt das Passwort des Imperia-Users nicht vor, wird der hier definierte LDAP-Account verwendet bzw. der *Anonymous Account*, falls hier keine Angaben gemacht wurden.

7.7.4.2.3 Liste der User-Container

Dies ist eine Liste der Container, in denen die möglichen User enthalten sind:

```
user_dn_1    ou=Entwicklung, o=Imperia, c=de
user_dn_2    ou=Verkauf, o=Imperia, c=de
user_dn_3    ou=Vertrieb, o=Imperia, c=de
```

Aus Effizienzgründen wird unterhalb der angegebenen Container die Suche nach Usern auf eine Ebene beschränkt. Im oben gezeigten Beispiel genügt es also nicht, folgendes anzugeben:

```
user_dn_1    o=Imperia, c=de
```

7.7.4.2.4 Liste der Attribute

Diese Liste enthält alle User-Attribute, die im Verzeichnisdienst abgespeichert werden. Hier werden auch die Namen festgelegt, unter denen die Attribute gespeichert werden.

```
USERFIELDS
id *          uid
login *       cn
name          sn
comment        description
fname          fn
password       userPassword
telnumber      telephonenumber
END_USERFIELDS
```

Die Attribute werden als Liste zwischen den Schlüsselwörtern USERFIELDS und END_USERFIELDS eingeschlossen. Die mit Sternchen gekennzeichneten Felder (login und uid) sind Pflichtfelder.

Weiterhin muss gewährleistet werden, dass die Textdarstellung des LDAP-Attributs für id nur numerisch/positiv ist und das Feld login innerhalb der Imperia-User eindeutig ist.

Neben der ID sind die in Imperia definierten Dateneingabefelder für den User als Attribute einstellbar. Diese Felder sind im Einzelnen:

name	comment	fname
login	password	language
homepage	addlinkmenu	addtextmenu
adddescmenu	addtexticon	bgcolor
street	city	zip
country	telnumber	cellular
faxnumber	background	email

Felder, die nicht im Verzeichnisdienst abgelegt werden können, werden im Dateisystem des Imperia-Servers abgelegt.

Das Feld password wird dabei von Imperia eigentlich nicht mehr benutzt, es sei denn, man verwendet anstelle des LDAP-Plug-Ins das ursprüngliche Passwort-Plug-In. In diesem Fall sollte man normalerweise auch noch die Passwort-Verschlüsselung einstellen (siehe oben).



Hinweis:

Bei dem Verzeichnisdienst von Microsoft® (ActiveDirectory) werden anstelle einer eindeutigen numerischen User- bzw. Rollen-ID die so genannten binären SIDs verwendet. Diese werden auch in Imperia akzeptiert, allerdings muss dafür das Gruppierungsschema auf ADIR gesetzt werden. Imperia setzt im Folgenden voraus, dass die jeweiligen RIDs der SIDs innerhalb der Menge der von Imperia akzeptierten User bzw. Rollen eindeutig sind. Dies ist im Falle einer einzelnen Windows® 2000 bzw. Windows® XP Domäne garantiert, andernfalls müssen Anpassungen vorgenommen werden.

7.7.4.25 Passwort-Verschlüsselung

Diese Einstellung beeinflusst nur das Standard-Authentifizierungs-Plug-In (Password):

```
password-encryption none
```

Zurzeit sind die Verfahren *none* und *MD5* implementiert.

7.7.4.26 Passwort-Kodierung

Für die Anmeldung (binding) an den LDAP-Server kann eine Kodierung des Passworts vor der Übertragung notwendig sein, da es nicht im Klartext übertragen werden darf.

```
password-encoding      SASL
```

Als implementierte Standardverfahren stehen *none* und *SASL* zur Verfügung. Für die Verwendung von *SASL* muss Version 3 des LDAP-Protokolls verwendet werden.

7.7.4.27 LDAP-Filter für gültige Imperia-User

Dieser Filter dient zur Identifizierung von gültigen, also bestehenden Imperia-Usern.

```
valid_user          (&(objectclass=Person) (uidnumber>=0))
```

Objekte (mit DIT), die diese erste Bedingung erfüllen, werden von Imperia als User anerkannt.

7.7.4.28 Liste der Rollen-Container

Dies ist eine Liste aller Container aus dem Verzeichnisdienst, in denen die möglichen Rollen enthalten sind:

```
group_dn_1          ou=Roles, o=Imperia, c=de
```

Hier können lediglich die Felder `id`, `name` und `comment` eingestellt werden. Das Feld `id` ist das einzige Pflichtfeld und muss in seiner Darstellung numerisch sein.



Hinweis:

Bei der Verwendung des Gruppierungsschemas ADIR gilt hierbei - wie bei den ID-Nummern der User - die gleiche Ausnahme.

7.7.4.29 Liste der Rollen-Attribute

Dies ist eine Liste aller Attribute der Rolle, die im Verzeichnisdienst gespeichert werden. Diese Liste muss analog zur Liste der User in Tags geklammert werden:

```
GROUPFIELDS
  id          gid
  name        name
  comment     description
END_GROUPFIELDS
```

7.7.4.30 LDAP-Filter zur Rollen-Identifikation

Dieser Filter dient zur Identifizierung der von Imperia verwalteten Rolle.

```
valid_group        (&(objectclass=Group) (gidnumber>=0))
```

Objekte, die diese Bedingung erfüllen, werden von Imperia als Rollen angesehen.

7.7.4.31 Gruppierungs-Schema

Welcher User welche Rolle annehmen kann und welche Rolle von welchen Usern angenommen werden kann, muss von Imperia auf irgendeine Weise einsehbar sein. Bei Verzeichnisdiensten gibt es keinen einheitlichen Standard. Imperia unterstützt zunächst das PAM-Schema.

grp_scheme PAM

Rollen beinhalten hierbei im Attribut `memberuid` (dieses Attribut ist *multivalued*) die `uid` der User, die ihnen angehören (**Achtung!** Nicht die `uidnumber`.). User hingegen besitzen kein Attribut, in dem alle Rollen aufgeführt sind, die sie annehmen können. Im User-Datensatz erscheint unter `gidnumber` die Hauptgruppe des Users, die zu den unter `memberuid` angegebenen hinzugefügt werden muss.

In einem späteren Schritt wird das Gruppierungsschema der Standard-Installation von ActiveDirectory (Windows® 2000) implementiert, die den Namen ADIR erhalten wird.

7.7.4.2.12 Superuser-Rolle

Die Superuser-Rolle aus Imperia (standardmäßig die Rolle mit der ID 0) kann bei Benutzung des LDAP-Plug-Ins zur Benutzerhaltung einer beliebigen Rolle zugewiesen werden.

superuser_group cn=cvs, ou=Roles, o=Imperia, c=de

Meldet sich ein User, der diese Rolle inne hat an, erhält er alle Rechte der Superuser-Rolle.

7.7.4.2.13 Superuser-Zugang



Hinweis:

Diese Einstellung beeinflusst nur das LDAP-Authentifizierungs-Plug-In.

Die Verwaltung des in Imperia angelegten Superusers kann nicht über einen im Verzeichnisdienst abgelegten User erfolgen. Trotzdem muss der Superuser auf einen im Verzeichnisdienst abgelegten User abgebildet werden. Dies geschieht mit Hilfe des folgenden Eintrags:

superuser cn=tgier, ou=People, o=Imperia, c=de

Meldet sich ein User als Superuser an, wird das eingegebene Passwort mit den Zugangsdaten des angegebenen DN-Eintrags verglichen und ein temporärer Benutzer mit der ID 1 angelegt.

7.7.4.2.14 Benutzeranbindung



Hinweis:

Diese Einstellung beeinflusst nur das LDAP-Authentifizierungs-Plug-In.

Ist diese Variable gesetzt, wird im LDAP-Authentifizierungs-Plug-In bei jedem CGI-Aufruf eine Anbindung des Benutzers an den Verzeichnisdienst vorgenommen.

userbind yes

Zum Verständnis: Wird diese Variable weggelassen oder auf 0 gesetzt, werden von Imperia alle Informationen mit Hilfe des konfigurierten Administrator-Zugangs aus dem Verzeichnisdienst abgeholt.

7.7.4.2.15 Schreibzugriff

Der Schreibzugriff auf den Verzeichnisdienst via LDAP wird durch Setzen folgender Variablen unterbunden:

readonly yes

7.7.4.3 Imperia als Frontend für einen LDAP-Verzeichnisdienst



Hinweis:

Diese Funktion ist zurzeit noch in der Vorbereitung.

Soll Imperia als Frontend für einen LDAP-kompatiblen Verzeichnisdienst verwendet werden, sind folgende Einstellungen notwendig:

7.7.4.3.1 RDN-Attribut eines neuen User-Eintrags

```
user_rdn_attribute      cn
```

7.7.4.3.2 Objektklassen eines neuen Imperia-Users

```
OBJECTCLASSES_USER
top
person
ImperiaUser
END_OBJECTCLASSES_USER
```

7.7.4.3.3 Das RDN-Attribut eines neuen Rollen-Eintrags

```
group_rdn_attribute      cn
```

7.7.4.3.4 Objektklassen einer neuen Imperia-Rolle

Objektklassen, zu denen eine neue von Imperia angelegte Gruppe gehört:

```
OBJECTCLASSES_GROUP
top
posixGroup
END_OBJECTCLASSES_GROUP
```

7.8 Authentifizierung über HTTP

Eine Alternative zur Benutzer-Authentifizierung über Imperia oder einen Verzeichnisdienst bietet das Basic-Plug-In. Es unterscheidet sich von den anderen Authentifizierungsmethoden in zweierlei Hinsicht:

- Die Authentifizierung geschieht ohne Cookie.
- Die Imperia-Authentifizierung wird außer Kraft gesetzt.

Stattdessen übernimmt der Webserver die Authentifizierung der Benutzer. Erfolgreich angemeldete Benutzer erhalten Zugriff auf das CMS.



Wichtig:

Passwörter können bei der HTTP-Authentifizierung nicht über den Menüpunkt Persönliche Einstellungen geändert werden.

Die Autorisierung angemeldeter Benutzer geschieht jedoch weiterhin über die Benutzer- und Rollenverwaltung des Systems. Ein Benutzer muss also in Imperia existieren, damit er das System den für ihn verfügbaren Rollen entsprechend voll nutzen kann.

Ohne Imperia-Userprofil hat ein authentifizierter Benutzer keine Rechte im System. Er hat dann lediglich Zugriff auf den Menüpunkt "Persönliche Einstellungen" und das interne Mail-System.

7.8.1 Konfiguration

Um die HTTP-Authentifizierung nutzen zu können, müssen Sie folgende Schritte ausführen:

1. Richten Sie das Verzeichnis `cgi-bin` in der Konfiguration Ihres Webservers als zugriffsgeschützten Bereich ein. Die hierfür notwendigen Schritte entnehmen Sie bitte der Dokumentation Ihres Webservers.



Hinweis:

Beachten Sie, dass Imperia die Autorisierung anhand der Benutzerdaten vornimmt, die vom Webserver übergeben werden. HTTP-Authentifizierungsdaten und Imperia-Benutzerprofile müssen also übereinstimmen, damit den Benutzern ihre jeweiligen Rechte korrekt zugewiesen werden können.

2. Die normale Imperia-Login-Seite (`htdocs/imperia/index.html`) sollte bei der HTTP-Authentifizierung umgangen werden. Diese Seite sollte also nicht mehr erreichbar sein. Stattdessen sollte direkt das Skript `site_main.pl` im Verzeichnis `cgi-bin` aufgerufen werden. Wenn Sie von anderen Seiten aus auf das CMS verweisen, müssen Sie diese Verweise bei der Umstellung auf die Authentifizierung per HTTP entsprechend anpassen. Sie können beispielsweise ein HTML-Dokument mit einem Verweis auf das Skript `site_main.pl` im `cgi-bin`-Verzeichnis erstellen. Benennen Sie die Datei entsprechend, dass sie automatisch aufgerufen wird, wenn jemand auf den Produktionsserver zugreift. Legen Sie dieses Dokument im Document-Root Ihres Produktionssystems ab.
3. Aktivieren Sie die HTTP-Authentifizierung in Imperia. Suchen Sie hierzu in der Datei `system.conf` im Verzeichnis `site/config` die Variable `AUTH_PLUGIN` und ändern Sie diese folgendermaßen:

```
"AUTH_PLUGIN" = "Basic"
```



Wichtig:

Es ist auch möglich, die HTTP-Authentifizierung des Webservers dem herkömmlichen Imperia-Login vorzuschenken. Diese Variante ist aber wenig sinnvoll, da Benutzer sich dann zusätzlich über die Imperia-Authentifizierung anmelden müssen. Ist dann das `cgi-bin`-Verzeichnis nicht passwortgeschützt, ist die Anmeldung am System sogar ganz unmöglich, da die Daten aus dem HTTP-Header nicht mit dem Webserver abgeglichen werden können.

Bei einem Zugriff auf `site_main.pl` erscheint nun die Anmeldeaufforderung des Webservers. Nach erfolgter Authentifizierung wird das Hauptmenü von Imperia angezeigt.

Kapitel 8. Workflows

Imperia bietet die Möglichkeit, den Produktionsprozess eines Dokuments an die bestehenden firmeninternen Prozesse anzupassen. Jeder einzelne Schritt im Arbeitsablauf, beispielsweise Anlegen, Editieren, Prüfen, und Freischalten eines Dokuments, wird durch einen Workflow-Schritt im Imperia-Workflow repräsentiert.

Aufbau eines Workflows

Ein Workflow besteht aus einem oder mehreren Workflow-Schritten, von denen jeder eine bestimmte Operation durchführt. Nur dazu autorisierte Benutzer können einen solchen Schritt ausführen. Alle benötigten Informationen zu einem Dokument sollten gesammelt werden, während es den Workflow durchläuft.

Denkbar wäre beispielsweise folgender Workflow: Im ersten Schritt werden die Meta-Informationen zu einem neuen Dokument eingegeben. Danach könnte man die Eingabe von Inhalten in dieses Dokument durch einen Redakteur als Workflow-Schritt definieren. Ein weiterer Schritt könnte eine Prüfung des Dokuments durch eine Prüfinstanz (Chefredakteur oder Rechtsabteilung) enthalten, in einem anderen könnte eine weitere Prüfung stattfinden (Mehr-Augen-Prinzip). Daran anschließen könnte sich ein Schritt, in dem wahlweise die Konvertierung des Dokuments in ein anderes Format (PDF, XML etc.) oder bei Eintreffen bestimmter Bedingungen dessen Löschung erfolgt.

Jeder Workflow hat mindestens einen Einstiegspunkt. Alternativ sind auch mehrere Einstiegspunkte möglich, so dass Dokumente aus verschiedenen Rubriken in einem Workflow unterschiedliche Wege durchlaufen können. Im Verlauf des Workflows können diese alternativen Abläufe zum Beispiel in einer Prüfinstanz wieder zusammenlaufen. Welchen der verfügbaren Einstiegspunkte ein Dokument nimmt, legen Sie bei der Erstellung von Rubriken fest.

Die einzelnen Schritte eines Workflows sind untereinander verbunden, wodurch sich jeder mögliche Weg, den ein Dokument darin gehen kann, abbilden lässt. Dazu gehören auch Verbindungen, die rückwärts führen, zum Beispiel dann, wenn die prüfende Instanz ein Dokument ablehnt, weil es fehlerhaft oder unvollständig ist. In so einem Fall geht das Dokument zur Korrektur zurück an eine bearbeitende Instanz.

Dokumente werden im Workflow völlig unabhängig voneinander behandelt, so dass mehrere Dokumente gleichzeitig den selben Schritt durchlaufen können.

Aufbau eines Workflow-Schrittes

Jeder Workflow-Schritt wird durch ein entsprechendes Workflow-Plug-In realisiert. Ihnen stehen eine Reihe von Standard-Plug-Ins zur Verfügung. Individuell angepasste Plug-Ins können Sie bei Bedarf erwerben oder entwickeln lassen.

Jeder Workflow-Schritt hat mindestens je einen Dateneingang sowie Datenausgang. Der Ausgang eines Workflow-Schritts wird mit einem Eingang des darauf folgenden Workflow-Schritts verbunden, woraus sich der Weg für die Dokumente durch den Workflow ergibt.

Ein Datenausgang, der nicht mit dem Eingang eines anderen Workflow-Schrittes verbunden ist, fungiert automatisch als Ende des Workflows bzw. Workflow-Zweigs. Dokumente, die einen solchen Schritt durchlaufen, verlassen anschließend den Workflow.

Vom Datenausgang eines Workflow-Schritts kann jedoch immer nur ein Weg weiterführen. Mehrere Ausgänge können durch Verwendung einer Weiche, an der das Dokument dupliziert wird, verwirklicht werden.

Änderungen am Workflow

Änderungen an einem Workflow betreffen nur Dokumente, die neu erzeugt oder aus dem Archiv re-importiert werden. Dokumente, die sich bereits vor der Änderung im Workflow befunden haben, sind nicht von den Änderungen betroffen.

Wenn Sie an Workflows für den Hermes-Daemon Änderungen vornehmen, ist ein Neustart des Daemons notwendig, um die Änderungen zu aktivieren. Dies trifft beispielsweise für automatisierte Import-Workflows zu.

8.1 Workflows verwalten

Sämtliche Funktionen für die Verwaltung von Workflows sind im Hauptmenü **Workflow** zusammengefasst. Hier können Sie Workflows neu angelegen, ändern und löschen. Ferner lassen sich Zugriffsrechte für jeden Workflow setzen oder entziehen. Im Einzelnen sind das folgende Rechte:

- Schreiben
- Ändern
- Lesen und
- Löschen

Zum Ändern von Dateirechten lesen Sie bitte auch Abschnitt 4.11 **Dateirechte in Imperia verwalten** auf Seite 75.

8.1.1 Workflow-Zugriffsrechte ändern

Workflow-Zugriffsrechte werden über den Menüpunkt **Ändern der Workflow-Zugriffsrechte** im Ordner **Struktur** geändert. Es öffnet sich folgender Dialog:

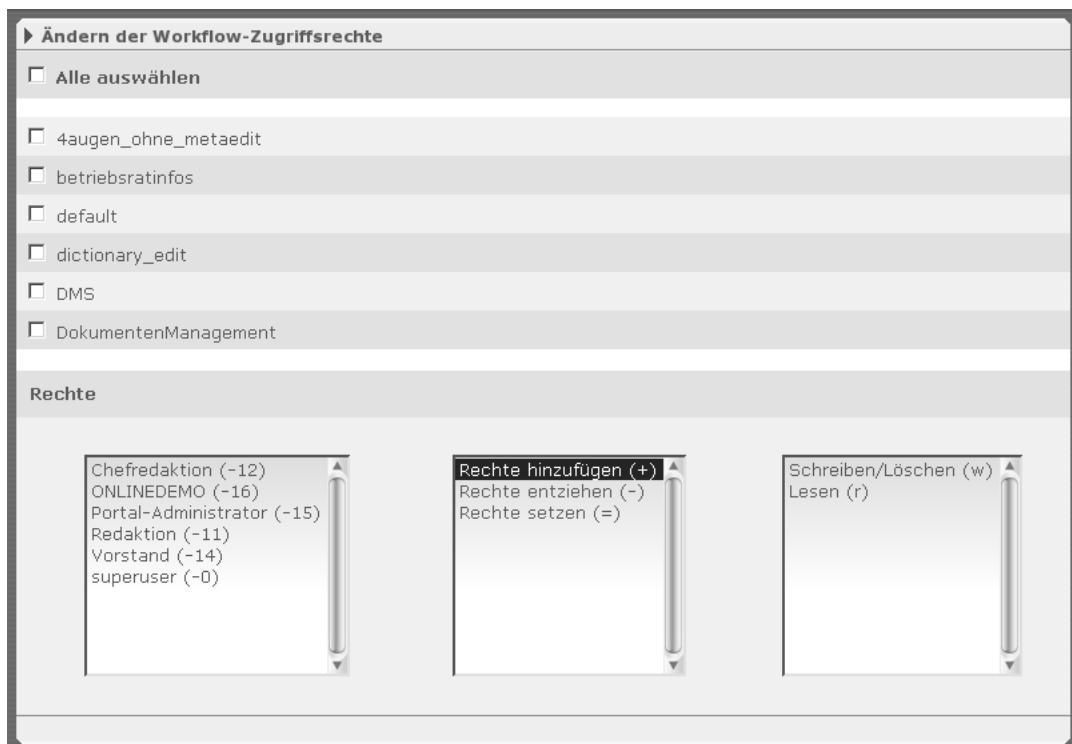


Abb. 8.1: Dialog Ändern der Workflow-Zugriffsrechte

Der Dialog ist in zwei Abschnitte gegliedert:

Dateiliste

Diese Liste enthält alle vorhandenen Workflows. Zu jedem Eintrag gibt es eine Checkbox, über die der entsprechende Workflow zum Ändern markiert wird. Die Checkbox in der blauen Titelleiste markiert gleichzeitig alle Einträge in der Liste.

Rechte

In diesem Abschnitt werden Schreib-, Lese- und Löschrechte für die in der Dateiliste markierten Workflows gesetzt. Lesen Sie hierzu Abschnitt 4.11 **Dateirechte in Imperia verwalten** auf Seite 75.

8.1.2 Neuen Workflow anlegen

Um einen Workflow anzulegen, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Workflow anlegen** im Ordner **Workflow**. Es wird folgender Dialog geöffnet:

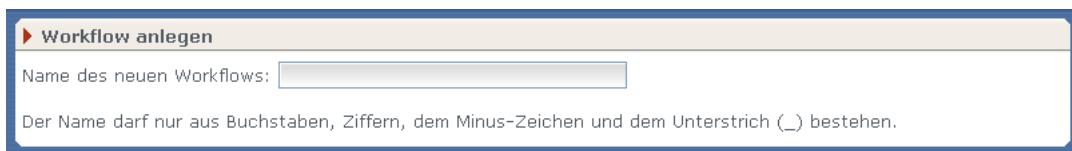


Abb. 8.2: Dialog Workflow anlegen

Tragen Sie den Namen des neuen Workflows ein. Der Name darf nur aus Buchstaben, Ziffern, dem Minus-Zeichen und dem Unterstrich (_) bestehen.

2. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs.

Es wird das Grid gestartet, in dem der neue Workflow gestaltet und gespeichert wird. Lesen Sie hierzu auch Abschnitt 8.2.3 **Bedienung des Grids** auf Seite 105.

8.1.3 Workflow bearbeiten

Workflows werden im Grid bearbeitet. Gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Workflow bearbeiten** im Ordner **Workflow**. Es wird eine Liste aller vorhandenen Workflows angezeigt.
2. Klicken Sie auf den Button **Bearbeiten** neben dem entsprechenden Workflow.
Es wird das Grid gestartet, in dem der gewählte Workflow bearbeitet werden kann.

Die Bedienung des Grids entnehmen Sie bitte Kapitel 8 **Workflows**.

8.1.4 Workflow löschen

Um einen Workflow zu löschen, gehen Sie wie folgt vor:

1. Klicken Sie auf den Menüpunkt **Workflow löschen** im Ordner **Workflow**. Es wird eine Liste aller vorhandenen Workflows angezeigt.
2. Klicken Sie auf den Button **Löschen** neben dem entsprechenden Workflow.
Es erscheint eine Sicherheitsabfrage, die Sie mit **Okay** bestätigen müssen. Anschließend wird der Workflow gelöscht. Der Button **Zurück** bricht die Aktion ab.

8.2 Das Grid

Beim Grid handelt es sich um eine Java-Applikation, mit deren Hilfe Workflows innerhalb von Imperia erstellt und verwaltet werden. Um mit dem Grid arbeiten zu können, muss auf dem Client-Rechner für den benutzten Browser das Java-Plug-In installiert werden.

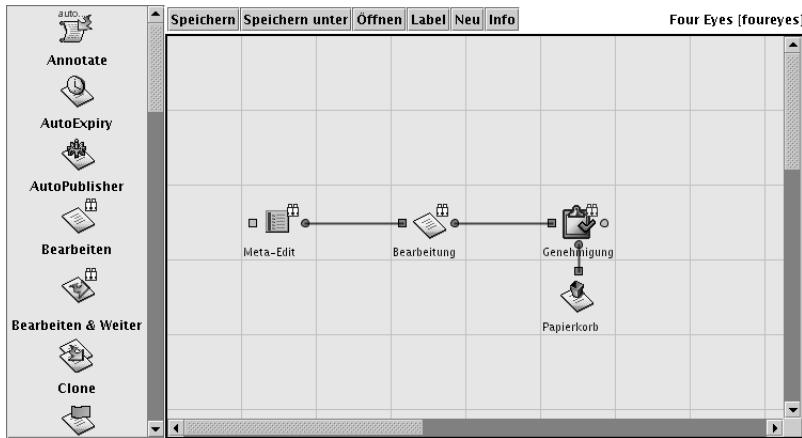


Abb. 8.3: Das Grid

Die Bedienungsoberfläche des Grids gliedert sich in folgende Bestandteile:

- Plug-In-Leiste,
- Button-Leiste und
- Arbeitsbereich.

8.2.1 Plug-In-Leiste

In der Plug-In-Leiste werden alle verfügbaren Plug-Ins durch Icons dargestellt.

8.2.2 Button-Leiste

In der Button-Leiste stehen verschiedene Buttons für Dateioperationen zur Verfügung. Weiterhin wird am rechten Rand der Titel des aktuell geladenen Workflows angezeigt. Die Buttons haben im einzelnen folgende Funktion:

Speichern

Speichert den geladenen Workflow und übernimmt Änderungen.

Speichern unter

Speichert den geladenen Workflow unter einem neuen Namen.

Öffnen

Öffnet die Liste aller eingerichteten Workflows, aus der dann ein Workflow zum Öffnen ausgewählt werden kann. Lesen Sie hierzu Abschnitt 8.1.3 **Workflow bearbeiten** auf Seite 104.

Neu

Legt einen neuen Workflow an. Es öffnet sich der Dialog **Workflow anlegen**. Lesen Sie hierzu Abschnitt 8.1.2 **Neuen Workflow anlegen** auf Seite 103.

Titel

Öffnet einen Dialog, in dem Titel und Beschreibung des aktuellen Workflow geändert werden können.

Info

Öffnet einen Dialog, in dem Copyright- und Versionsinformationen angezeigt werden.

8.2.3 Bedienung des Grids

Zur Bedienung des Grids wird vor allen Dingen die Maus verwendet. Eine Bedienung ausschließlich über die Tastatur ist nicht möglich.

Neue Workflow-Schritte binden Sie in den Workflow ein, indem Sie das entsprechende Icon aus der Plug-In-Leiste per Drag-and-Drop im Arbeitsbereich des Grids positionieren. Lesen Sie hierzu Abschnitt 8.2.3.2 **Workflow-Schritt anlegen** auf Seite 106. Jedes Feld des Arbeitsbereichs kann genau einen Workflow-Schritt aufnehmen. Bereits vorhandene Workflow-Schritte markieren Sie mit einem Klick der linken Maustaste. Markierte Workflow-Schritte sind orange unterlegt. Das gleichzeitige Markieren mehrerer Felder ist nicht möglich.

Bei konfigurierbaren Workflow-Schritten öffnen Sie per Doppelklick einen Dialog, in dem Sie die festgelegten Eigenschaften zu diesem Workflow-Schritt konfigurieren können. Lesen Sie hierzu auch Abschnitt 8.2.3.3 **Eigenschaften eines Workflow-Schrittes einstellen** auf Seite 106. Ein Rechtsklick auf einen beliebigen im Arbeitsbereich des Grids positionierten Workflow-Schritt öffnet das Kontextmenü mit den Befehlen **Bearbeiten**, **Duplizieren** und **Löschen**.

8.2.3.1 Workflow-Plug-Ins

Workflow-Schritte werden durch die Workflow-Plug-Ins repräsentiert, die Sie in der Plug-In-Leiste des Grids sehen.

Bei den Workflow-Schritten unterscheidet man grundsätzlich zwischen interaktiven und nicht interaktiven Plug-Ins. Interaktive Plug-Ins benötigen beim Durchlauf des Workflows eine Eingabe des Benutzers. Ein Beispiel hierfür ist das Bearbeiten-Plug-In. Nicht interaktive Plug-Ins laufen automatisiert ab und benötigen daher keine Eingabe durch den Benutzer. Ein Beispiel hierfür ist der AutoPublisher. Zur besseren Unterscheidung sind die Icons interaktiver Workflow-Plug-Ins mit zwei Figuren ☺ kenntlich gemacht.

Außerdem sind einige Plug-Ins konfigurierbar bzw. müssen Sie ihre Konfiguration auf den jeweiligen Workflow anpassen, damit Sie sie verwenden können. Bei den interaktiven Plug-Ins handelt es sich hierbei meistens um die Einstellung der Zugriffsberechtigungen. Nicht interaktive Plug-Ins haben sehr unterschiedliche Einstellungsmöglichkeiten. Lesen Sie hierzu die Beschreibung des jeweiligen Plug-Ins in Abschnitt 8.5 **Workflow-Plug-Ins** auf Seite 109.

8.2.3.2 Workflow-Schritt anlegen

Um einen Workflow-Schritt dem Workflow hinzuzufügen, wird eine Instanz eines Plug-Ins im Arbeitsbereich erstellt:

- Ziehen Sie per Drag & Drop das Plug-In, das Sie verwenden wollen, aus der Plug-In-Leiste in den Arbeitsbereich und positionieren Sie es in einem freien Feld. Es wird eine Instanz des Plug-Ins, und somit ein Workflow-Schritt, erzeugt.
- Anschließend müssen Sie das neue Plug-In mit anderen Workflow-Schritten verbinden. Lesen Sie hierzu Abschnitt 8.2.3.7 **Workflow-Schritte verbinden** auf Seite 107.
- Einige Plug-Ins müssen außerdem noch konfiguriert werden. Lesen Sie hierzu den folgenden Abschnitt sowie die Beschreibung der einzelnen Plug-Ins in Abschnitt 8.5 **Workflow-Plug-Ins** auf Seite 109.

Je nach Aufbau eines Workflows kann ein Plug-In mehrfach darin vorkommen.

8.2.3.3 Eigenschaften eines Workflow-Schrittes einstellen

Um die Eigenschaften eines Workflow-Schrittes einzustellen, gehen Sie wie folgt vor:

- Doppelklicken Sie im Arbeitsbereich des Grids auf den Workflow-Schritt, dessen Eigenschaften Sie einstellen wollen.

Es wird der Dialog **Workflow-Schritt bearbeiten** geöffnet. Dieser Dialog kann auch über das Kontext-Menü der rechten Maustaste geöffnet werden (Option **Bearbeiten**). Die Einstellungsmöglichkeiten in dem Eigenschaften-Dialog sind für die einzelnen Plug-Ins unterschiedlich. Die folgende Abbildung zeigt exemplarisch das Dialogfenster für das Edit-Plug-In:

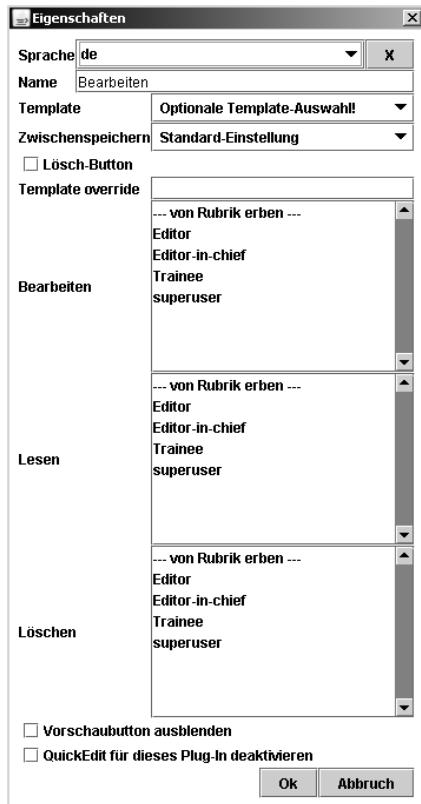


Abb. 8.4: Dialog Workflow-Schritt Eigenschaften (Edit-Plug-In)

Einige Plug-Ins benötigen keine weitere Konfiguration. Lesen Sie mehr über die Konfigurationsmöglichkeiten der einzelnen Plug-Ins im Abschnitt 8.5 **Workflow-Plug-Ins** auf Seite 109

8.2.3.4 Workflow-Schritt im Arbeitsbereich verschieben

Workflow-Schritte werden innerhalb des Arbeitsbereichs mit der Maus verschoben. Klicken Sie den entsprechenden Workflow-Schritt an, halten Sie die Maustaste gedrückt und verschieben Sie den Workflow-Schritt in ein anderes freies Feld. Bereits existierende Verbindungen bleiben erhalten.

8.2.3.5 Workflow-Schritt duplizieren

Wird beim Verschieben gleichzeitig die Strg-Taste gedrückt gehalten, wird der markierte Workflow-Schritt dupliziert. Er kann dann beliebig innerhalb des Arbeitsbereiches verschoben und neu positioniert werden. Eventuell vorhandene Verbindungen des ursprünglichen Workflow-Schrittes werden nicht dupliziert.

8.2.3.6 Workflow-Schritt löschen

Markierte Workflow-Schritte werden mit Hilfe der Entf-Taste oder mit Hilfe des Kontextmenüs der rechten Maustaste gelöscht. Wird ein Workflow-Schritt gelöscht, werden auch die bestehenden Verbindungen zu anderen Workflow-Schritten entfernt.

8.2.3.7 Workflow-Schritte verbinden

Das Verbinden der einzelnen Workflow-Schritte untereinander erzeugt die Wege, die ein Dokument im Workflow gehen kann. Workflow-Schritte werden mit Hilfe der Maus untereinander verbunden. Ausgehend vom Datenausgang eines Workflow-Schrittes wird mit gedrückter linker Maustaste eine Verbindung, symbolisiert durch eine schwarze Linie, zu einem Dateneingang des folgenden Workflow-Schrittes gezogen. Wird die Maustaste an einem Dateneingang losgelassen, ist die Verbindung etabliert.

8.2.3.8 Verbindungen lösen

Falls eine falsche Verbindung etabliert wurde, kann diese entweder gelöscht oder durch eine korrekte Verbindung ersetzt werden.

Zum Löschen klickt man mit der rechten Maustaste auf die Verbindung und wählt im Kontext-Menü den Eintrag **Diese Verbindung löschen**.

Um eine Verbindung zu ersetzen, muss eine neue Verbindung ausgehend vom gleichen Datenausgang, den auch die falsche Verbindung verwendet, zum korrekten Dateneingang gezogen werden. Beim Loslassen der Maustaste wird die alte Verbindung durch die neue ersetzt.

8.2.4 Zugriffsrechte für Workflow-Schritte

Für viele Workflow-Schritte kann festgelegt werden, welche Rollen zum Zugriff auf diesen Schritt berechtigen.

Alle User, die eine der Rollen annehmen können, erhalten dann über ihren Schreibtisch Zugriff auf die Dokumente, für die der jeweilige Workflow-Schritt ausgeführt werden soll, und zwar dann, wenn das Dokument diesen Workflow-Schritt im Workflow erreicht.

Auf den Schreibtisch der User, die diesen Schritt nicht ausführen dürfen, werden diese Dokumente nicht angezeigt. Für jeden Workflow-Schritt muss mindestens eine Rolle zum Zugriff berechtigen. Andernfalls laufen die Dokumente an diesem Workflow-Schritt auf, ohne dass sie bearbeitet werden können.

In der Praxis sieht ein User, der aufgrund seiner aktuellen Rolle Inhalt in Dokumente eingeben darf, ein Dokument also erst dann, wenn der vorangegangene Workflow-Schritt, zum Beispiel Meta-Informationen eingeben, abgeschlossen ist und das Dokument den nächsten Workflow-Schritt, zum Beispiel Inhalt eingeben, erreicht.

Hat der User nur Zugriff auf den Workflow-Schritt Inhalt eingeben, erscheint das Dokument, sobald er diesen Workflow-Schritt abschließt, nicht mehr auf seinem Schreibtisch. Es erscheint dann auf dem Schreibtisch der User, die den nächsten Workflow-Schritt, zum Beispiel Prüfung, ausführen dürfen.

8.2.4.1 Zugriffsrechte einstellen

Um die Zugriffsrechte für einen Workflow-Schritt einzustellen, gehen Sie wie folgt vor:

1. Klicken Sie im Arbeitsbereich des Grids doppelt auf den Workflow-Schritt, dessen Eigenschaften Sie einstellen wollen.

Es wird der Dialog **Workflow-Schritt bearbeiten** geöffnet. Dieser Dialog kann auch über das Kontext-Menü der rechten Maustaste geöffnet werden.

Es gibt drei Stufen von Zugriffsrechten auf einen Arbeitsschritt:

- Bearbeiten
- Lesen
- Löschen

Bearbeiten

Eine Rolle, die Zugriff auf einen Workflow-Schritt mit der Stufe **Bearbeiten** erlaubt, berechtigt zum Bearbeiten von Dokumenten, die diesen Workflow-Schritt erreichen.

Lesen

Eine Rolle, die Zugriff auf einen Workflow-Schritt mit der Stufe **Lesen** erlaubt, berechtigt zum Lesen von Dokumenten, die diesen Workflow-Schritt erreichen.

Löschen

Eine Rolle, die Zugriff auf einen Workflow-Schritt mit der Stufe **Löschen** erlaubt, berechtigt zum Löschen von Dokumenten, die diesen Workflow-Schritt erreichen.

2. Markieren Sie in den entsprechenden Abschnitten eine oder mehrere Rollen, die auf den gewählten Workflow-Schritt Zugriff der entsprechenden Stufe bekommen sollen.

Mehrere Rollen werden mit Hilfe der Strg-Taste markiert.

Über den Eintrag **von Rubrik erben** werden die Rechte aus der mit diesem Workflow verknüpften Rubrik übernommen. Die in der Rubrik für einzelne Rollen konfigurierten Lese-, Schreib- und Löschrechte gelten dann auch für das entsprechende Workflow-Plug-In.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um die Einstellungen zu speichern. Der Button **Abbrechen** schließt den Dialog, ohne Änderungen zu speichern.

8.2.42 Zugriffsrechte entfernen

Um Zugriffsrechte auf einen Workflow-Schritt zu entfernen, gehen Sie wie folgt vor:

1. Klicken Sie im Arbeitsbereich des Grids doppelt auf den Workflow-Schritt, dessen Eigenschaften Sie einstellen wollen.

Es wird der Dialog **Workflow-Schritt bearbeiten** geöffnet. Dieser Dialog kann auch über das Kontext-Menü der rechten Maustaste geöffnet werden.

2. Entfernen Sie in den entsprechenden Abschnitten die Markierung der Rollen, denen in Zukunft kein Zugriff auf den gewählten Workflow-Schritt gestatten werden soll.

Klicken Sie dazu mit gedrückter Strg-Taste auf einen markierten Eintrag. Die Markierung wird daraufhin aufgehoben und die entsprechende Rolle hat in Zukunft kein Zugriffsrecht mehr.

3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um die Einstellungen zu speichern. Mit dem Button **Abbrechen** schließen Sie den Dialog, ohne Änderungen zu speichern.

8.3 Workflow bearbeiten

Um einen vorhandenen Workflow zu bearbeiten, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Workflow** auf den Menüpunkt **Workflow bearbeiten**.

Es wird die Workflow-Liste geöffnet, in der alle im System eingerichteten Workflows angezeigt werden.

2. Wählen Sie den Workflow, den Sie bearbeiten möchten, durch einen Klick auf den entsprechenden Button **Bearbeiten** aus.

Es wird das Grid (siehe Seite 104) gestartet und der Workflow dargestellt.

8.4 Workflow löschen

Um einen bestehenden Workflow zu löschen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Workflow** auf den Menüpunkt **Workflow löschen**.

Es wird die Workflow-Liste geöffnet.

2. Klicken Sie in der Liste auf den Namen des Workflows, der gelöscht werden soll.

Der Workflow wird ohne weitere Sicherheitsabfrage gelöscht und das Hauptmenü wird wieder geöffnet.

Das Löschen eines Workflows ist damit abgeschlossen.

8.5 Workflow-Plug-Ins

Im Folgenden finden Sie eine Liste aller Workflow-Plug-Ins, die mit Imperia 8 ausgeliefert werden.

Für die meisten Plug-Ins können Sie einstellen, welche Rollen zum Zugriff berechtigt sind. Lesen Sie hierzu Abschnitt 8.2.4 **Zugriffsrechte für Workflow-Schritte** auf Seite 108. Erläuterungen zu den einzelnen Plug-Ins und ihrer Konfiguration finden Sie jeweils in einem eigenen Abschnitt.

Grundsätzlich gibt es drei Kategorien von Plug-Ins:

- interaktive Plug-Ins, die eine Benutzereingabe erwarten,
- nicht interaktive Plug-Ins, die keine Benutzereingabe benötigen,

- semi-interaktive Plug-Ins, die unter manchen Voraussetzungen eine Benutzereingabe benötigen und unter manchen nicht.

Anhand des Icons können Sie erkennen, ob ein Plugin interaktiv, nicht interaktiv oder semi-interaktiv ist. Interaktive Plugins werden durch zwei weiß hinterlegte Figuren ergänzt; semi-interaktive durch eine einzelne und nicht interaktive zeigen Figur in ihrem Icon.

Das Bearbeiten-Plug-In ist ein typisches Beispiel für ein interaktives Plug-In. Es erwartet, dass ein User das Dokument editiert. Die Grid-Icons dieser Plug-Ins werden durch zwei weiße Figuren ergänzt.

Ein Beispiel für ein nicht interaktives Plug-In ist das Auto-Expiry-Plug-In, das keinerlei Usereingabe erwartet, sondern für jedes Dokument einen Zeitpunkt definiert, an dem es gelöscht werden soll.

Semi-interaktive Plug-Ins finden sich im Zusammenhang mit der Erstellung von Asset-Dokumenten. Beispielsweise das Plug-In Asset-Aktionen erfordert nur beim Upload eines einzelnen Assets eine Benutzereingabe. Beim Mehrfachupload erfolgt die Festlegung verfügbarer Assets hingegen automatisch.

8.5.1 Asset-Aktionen

Dieses Plug-In ist nur in Upload-Workflows für Asset-Dateien anwendbar. In diesem Workflow-Schritt legen Sie fest, welche Aktionen für ein Asset verfügbar sein sollen. Es handelt sich um ein semi-interaktives Plug-In, das nur beim Upload einzelner Assets eine Eingabe vom Benutzer erfordert. Beim Mehrfachupload bekommen alle hochgeladenen Assets automatisch alle auswählbaren Aktionen zugewiesen.

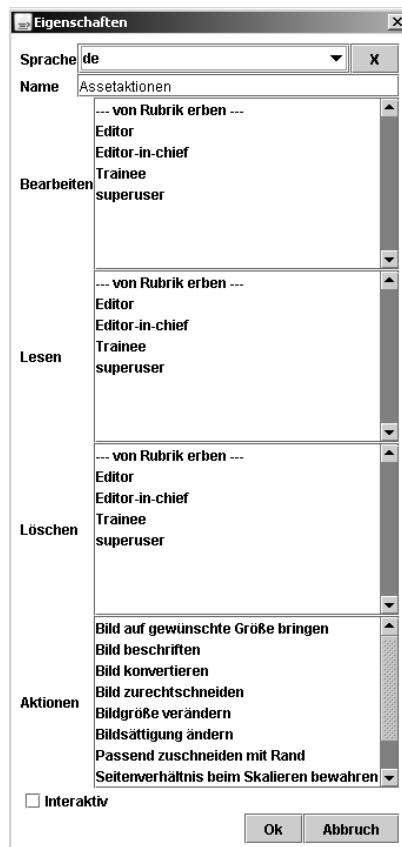


Abb. 8.5: Konfiguration des Plug-Ins Asset-Aktionen

Folgende Konfigurationsmöglichkeiten stehen zur Verfügung:

Bearbeiten

Über eine Auswahlliste können Sie festlegen, welche Rollen ein Asset-Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahlliste können Sie bestimmen, welche Rollen ein Asset-Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Asset-Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

Aktionen

Markieren Sie die Aktionen, die in dem entsprechenden Workflow-Schritt auswählbar sein sollen.

Interaktiv

Wird der Haken gesetzt, kann der Uploader aus den in der Plugin-Konfiguration ausgewählten Aktionen eine erneute Auswahl treffen. Wird der Haken nicht gesetzt, wird dem Uploader keine Auswahl angezeigt und alle hier ausgewählten Aktionen für die Bearbeitung des Assets stehen im MAM zur Verfügung.

8.5.2 Annotate

Dieses Plug-In erzeugt aus einem beliebigen Text automatisch einen History-Eintrag. Jedes Dokument, das vom Plug-In bearbeitet wird, erhält den gleichen History-Eintrag. Den Text des History-Eintrags legen Sie in der Plug-In-Konfiguration im Feld **History-Eintrag** fest.

8.5.3 Archiv-Bereiniger

Imperia archiviert bei einem Durchlauf des Workflows den jeweiligen Stand des Dokuments bei jedem Schritt. Es wird also für jeden Workflow-Schritt eine Archiv-Version des Dokuments erzeugt. Je umfangreicher ein Workflow ist, desto mehr archivierte Dokument-Versionen fallen an. Auf diese Weise kann das Archiv schnell sehr umfangreich werden, so dass es viel Speicherplatz belegt, unübersichtlich wird und seine Performance sich mehr und mehr verlangsamt.

Abhilfe kann hier der Archiv-Bereiniger schaffen. Mit diesem Plug-In können Sie bei jedem Durchgang des Workflows das Archiv automatisch von nicht beendeten Versionen des Dokuments bereinigen. Für die Konfiguration stehen Ihnen folgende Optionen zur Verfügung:

Aktuelle nicht abgeschlossene Versionen

Wenn Sie diese Option auswählen, werden alle nicht abgeschlossenen Versionen des Dokuments gelöscht, die vor der letzten beendeten Version erstellt wurden. Danach erstellte unbeendete Versionen bleiben erhalten.

Alle nicht abgeschlossenen Versionen

Bei dieser Option werden auch die unbeendeten Archiv-Versionen gelöscht, die nach der letzten beendeten Version erstellt wurden.

Alte nicht abgeschlossene Versionen

Wählen Sie diese Option, um nur die unbeendeten Versionen des Dokuments aus dem Archiv zu löschen, die vor der vorletzten beendeten Version erstellt wurden.

8.5.4 Auto-Expiry

Dieses Plug-In verleiht einem Dokument automatisch das aktuelle Datum als Ablaufdatum. Es wird also nach Beendigung des Workflows sofort gelöscht. Dafür muss auf dem Produktionssystem der Hintergrund-Daemon laufen. Dieses Plug-In benötigt keine weitere Konfiguration.

8.5.5 Auto-Publisher

Dieses Plug-In verleiht einem Dokument automatisch das aktuelle Datum als Freischaltdatum. Es wird also nach Beendigung des Workflows sofort auf das Zielsystem freigeschaltet. Dafür muss auf dem Produktionssystem der Hintergrund-Daemon laufen. Dieses Plug-In benötigt keine weitere Konfiguration.

8.5.6 Bearbeiten

Dieses Plug-In öffnet das Template zur Eingabe von Inhalten für den Benutzer. Eingegebene Daten können abgespeichert oder verworfen werden.

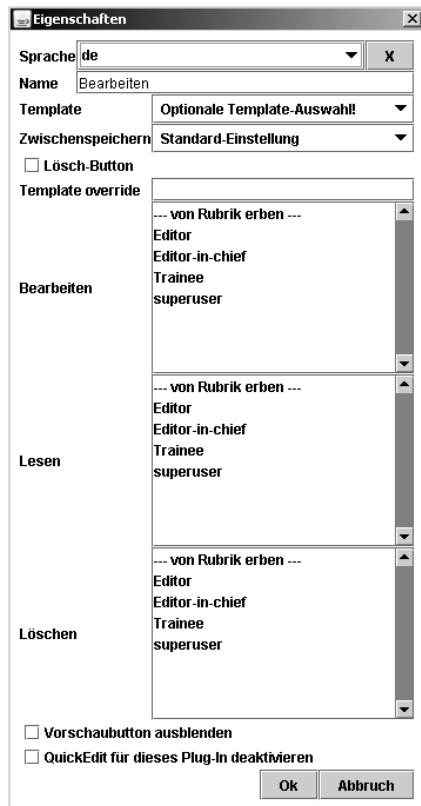


Abb. 8.6: Konfiguration des Bearbeiten-Plug-Ins

Folgende Konfigurationsmöglichkeiten stehen zur Verfügung:

Template

Über diese Auswahlliste können Sie festlegen, dass für diesen Schritt ein anderes Template verwendet werden soll als in den Rubrikeinstellungen definiert wurde.

Zwischenspeichern

Über diese Auswahliste können Sie eine Schaltfläche im Template ein- oder ausblenden, welche es erlaubt das Dokument zu speichern ohne zum Schreibtisch zurück zu wechseln.

Lösch-Button

Über diese Schaltfläche können Sie das geöffnete Dokument einschließlich Archiv und zugehörige Objekte löschen.

Template Override

Diese Option dient Ihnen dazu, ein vorausgewähltes Template durch ein anderes zu ersetzen. Sie können hier auch XX-Variablen verwenden, um beispielsweise einen Templatenamen mit einem variablen Suffix zu ergänzen.

Bearbeiten

Über eine Auswahliste können Sie festlegen, welche Rollen ein Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahliste können Sie bestimmen, welche Rollen ein Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

Vorschaubutton ausblenden

Blendet die Schaltfläche zur Vorschau eines Dokumentes im Template aus.

QuickEdit für dieses Plug-In deaktivieren

Ist diese Option gesetzt, wird das betroffene Plugin bei der Bearbeitung mittels der One-Click-Edit Toolbar übersprungen.

8.5.7 Bearbeiten und Weiterleiten

Dieses Plug-In ist eine Erweiterung des Bearbeiten-Plug-Ins. Es bietet einen zusätzlichen Ausgang (Süden), der verwendet wird, wenn ein Dokument gespeichert wird. Die Funktion des Ausgangs im Osten ist identisch mit der Funktion des östlichen Ausgangs des Bearbeiten-Plug-Ins.

Das „Bearbeiten und Weiterleiten“-Plug-In kann beispielsweise verwendet werden, um den Prozess zum Erzeugen von Inhalt statistisch zu erfassen.

Beispiel:

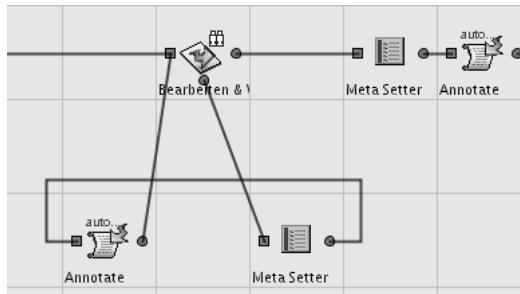


Abb. 8.7: Beispiel Bearbeiten & Weiter-Plug-In

Dieser Ausschnitt aus einem Workflow bildet folgende Funktion ab:

Jedesmal, wenn ein Dokument bearbeitet und gespeichert wird, verlässt es den Workflow-Schritt durch den südlichen Ausgang und durchläuft dann das MetaSetter-Plug-In und anschließend das Annotate-Plug-In.

Danach wartet es auf die nächste Bearbeitung. Wird der Workflow-Schritt abgeschlossen, verlässt das Dokument den Workflow-Schritt durch den östlichen Ausgang, durchläuft ebenfalls diese beiden Plug-Ins (damit die gleichen Aktionen wie beim Speichern durchgeführt werden) und gelangt anschließend in den nächsten Workflow-Schritt.

Die Konfiguration entspricht der des Bearbeiten-Plug-Ins.

8.5.8 Cat2DocMeta

Dieses Plug-In vereinfacht die Übertragung von Rubrik-Metainformationen in Dokumente. Tragen Sie in der Plug-In-Konfiguration die Namen der Rubrik-Metafelder ein, die Sie in das Dokument übertragen wollen.

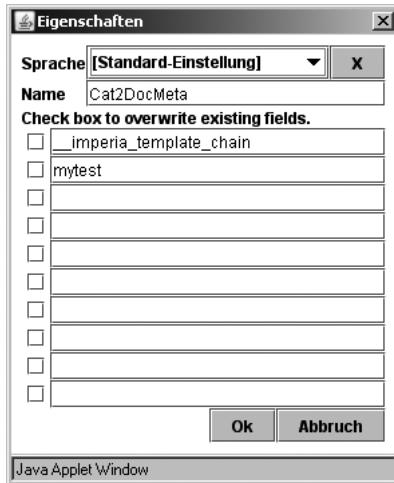


Abb. 8.8: Konfiguration des Plug-Ins Cat2DocMeta

8.5.9 Clone

Das Clone Workflow-Plug-In erzeugt eine (nahezu) identische Kopie eines Imperia Dokumentes. Anders als bei Imperia Copy Pages ist das neue Dokument ein eigenständiges Dokument und somit komplett unabhängig. Veränderungen am Eltern-Dokumentes haben somit keinen Einfluss auf das neu erzeugte Kind-Dokument. Es besteht jedoch eine "lockere" Beziehung zwischen den Dokumenten, welche den Umgang erleichtert. Es besteht die Möglichkeit über die Meta-Variablen `__imperia_parent` bzw. `__imperia_children` die Kind- respektive Eltern-Dokumente auszulesen.

Grundsätzlich werden beim Erstellen des Kind-Dokumentes alle Meta-Variablen des Eltern-Dokumentes übernommen. Diese können jedoch nach Belieben geändert oder auch gelöscht werden. Dabei ist wichtig zu beachten, dass alle durchgeführten Änderungen in genau der Reihenfolge durchgeführt werden, in welcher sie in der Plug-In Konfiguration auftauchen. Zudem werden Änderungen immer auf Basis der aktuell gültigen Meta-Variablen und deren Werten durchgeführt. Wird beispielsweise die Meta-Variable `directory` gelöscht, kann sie später nicht mehr referenziert werden.

Das neu erzeugte Dokument unterscheidet sich in den folgenden Meta-Variablen von dem Elter-Dokument:

- **`__imperia_node_id`**: Die Meta-Variable `__imperia_node_id` enthält die Node-Id des neu erzeugten Dokumentes und nicht die des Elter-Dokumentes.
- **`__imperia_created`, `__imperia_modified`**: Beide Meta-Variablen enthalten die Zeit in Sekunden für den Zeitraum seit dem 1 Januar 1904 GMT 00:00:00 für MacOS bzw 1 Januar 1970 GMT 00:00:00 für andere Betriebssysteme.
- **`__imperia_uid`**: Enthält die User-Id des aktuell angemeldeten Benutzers.
- **`__imperia_meta_expand`**: Ist die Meta-Variable `directory` vorhanden, enthält die Meta-Variable `__imperia_meta_expand` den Wert 1, welcher durch das MetaEdit-Plug-In gesetzt wurde. Wenn vorhanden und nicht Null wird die Meta-Datei-Direktive `<!--XX-METAMODE-->` true.
- **`__imperia_parent`**: Diese Meta-Variable enthält die Node-Id des Eltern-Dokumentes.
- **`__imperia_children`**: Diese Meta-Variable enthält alle Node-Ids der vorhandenen Kind-Dokumente. Beachten Sie, dass diese Variable nur im Eltern-Dokument vorhanden ist.

Zudem stehen die folgenden Konfigurationsmöglichkeiten zur Verfügung:

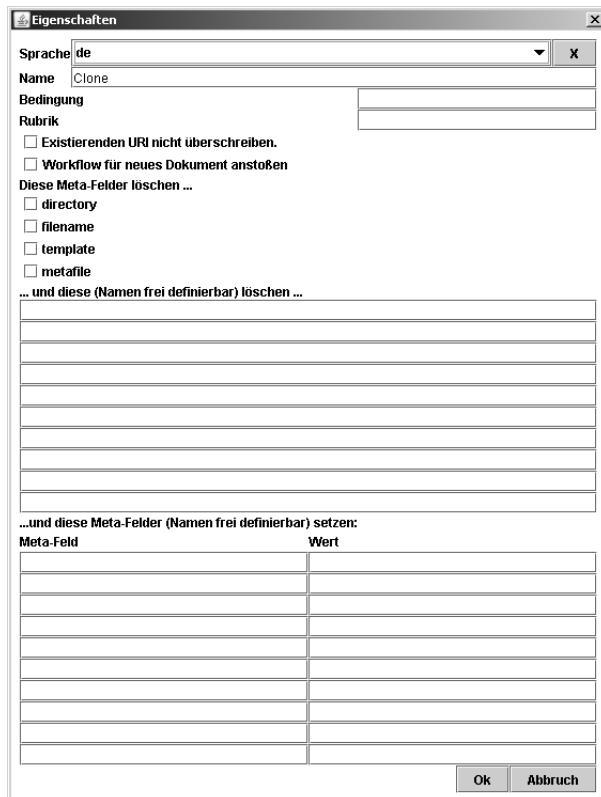


Abb. 8.9: Konfiguration des Plug-Ins Clone

Bedingung

Ein Dokument kann niemals ohne eine Bedingung erzeugt werden, um Endlosschleifen im Workflow zu verhindern. Es muss also eine Bedingung angegeben werden, die den Wert einer Variable überprüft um den Clone-Prozess starten zu können. Die Bedingung wird von der Imperia Template Engine als IF Anweisung interpretiert und muss den gängigen Syntaxregeln genügen.

Beispiel:

Wenn sie die Bedingung

`"<!--XX-create_copy-->" EQ "yes"`

eingeben, wird der folgenden Template Code evaluiert:

```
#IF ("<!--XX-create-copy-->" EQ "yes")
  Dokument wird geclont...
#ELSE
  Dokument wird nicht geclont...
#endif
```

Das Plug-In überprüft nicht den Inhalt der Bedingung. Sind Sie sich sicher, keine Endlosschleife zu erzeugen, können die eine Bedingung angeben, welche immer "true" ergibt (Beispiel: "1" oder "*Ich weiss was ich tue*").

Rubrik

Wie jedes Imperia Dokument muss auch ein geklontes Dokument in einer Rubrik gespeichert werden. Wird im Feld Rubrik nichts angegeben, speichert Imperia das Dokument in der Rubrik des Eltern-Dokumentes. Andernfalls muss eine Rubriken-Id der Form "/1/10" angegeben werden. Anstatt der Rubriken-Id kann auch der Rubrikenname angegeben werden. Dies hat jedoch einen negativen Einfluss auf die Performance, da zunächst nach dem Namen der Rubrik gesucht werden muss, was je nach Größe des Rubrikenbaums lange dauern kann. Zudem kann es zu Problemen kommen, wenn zwei Rubriken mit dem gleichen Namen existieren.

Existierende URI nicht überschreiben

Ist diese Option ausgewählt, versucht Imperia bereits bestehende Dokumente nicht zu überschreiben.

Workflow für neues Dokument anstoßen

Im Normalfall wird das neue Dokument erzeugt und danach nicht weiter bearbeitet. Ist diese Option jedoch gewählt, wird das Dokument durch den Workflow "geschoben" und stoppt erst vor dem ersten Workflow-Plug-In welches Benutzerinteraktion erfordert. (solche Workflow-Plug-Ins sind durch zwei kleine Figuren in der Rechten oberen Iconecke gekennzeichnet)

Diese Metafelder löschen ...

Die Meta-Variablen `directory,filename,template` und `metafile` können mit Informationen, welche aus der Rubrikendefinition ausgelesen werden initialisiert werden. Dies geschieht durch das wählen der jeweiligen Option. Werden diese Optionen nicht gewählt, werden die Werte aus dem Eltern-Dokument übernommen. Wird die Meta-Variable `directory` gelöscht, wird ebenfalls `__imperia_meta_expand` gelöscht (Details zu dieser Variable wurden bereits zuvor erläutert).

... und diese (Namen frei definierbar) löschen

Hier können bis zu zehn frei wählbare Meta-Variablen angegeben werden, welche ebenfalls aus dem neuen Dokument gelöscht werden sollen. Hinweis: An dieser Stelle können Imperia Template Variablen verwendet werden.

... und diese Meta-Felder (Namen frei definierbar) setzen:

Hier können bis zu zehn frei wählbare Meta-Variablen und dessen Werte angelegt werden. Existieren diese Variablen bereits werden sie überschrieben. Hinweis: An dieser Stelle können Imperia Template Variablen verwendet werden.

8.5.10 DocSelector

Das DocSelector-Plug-In hilft Ihnen bei der Behandlung von URI-Konflikten eines Imperia-Dokuments.

Sie können mit Imperia eine beliebige Anzahl von Dokumenten mit gleichem URI anlegen. Allerdings kann sich der URI eines Dokumentes ändern, wann immer es den Workflow durchläuft.

Wenn Sie mehrere Dokumente mit gleichem URI anlegen, ergibt sich daraus, dass im Document-Root eine Datei überschrieben wird. Oft ist dieser Effekt erwünscht. Manchmal ergibt sich daraus jedoch eine Beeinträchtigung für den Administrator, der den tatsächlichen Inhalt der Datei überwachen muss.

Mit dem DocSelector-Plug-In können Sie die Behandlung von URI-Konflikten an die spezifischen Anforderungen Ihrer Site anpassen. Dies ist auf Rubrikenebene möglich.

Jedesmal, wenn ein Dokument den Workflow-Schritt mit dem Plug-In durchläuft, wird dessen Rubrik nach Dokumenten mit dem selben URI durchsucht. Dabei können folgende Fälle eintreten:

- Es ist kein Dokument mit dem selben URI vorhanden. Nichts passiert, der Workflow läuft ganz normal weiter.
- Es existieren ein oder mehrere Dokumente mit dem selben URI. Sie können den Konflikt dann interaktiv auflösen. Für den Vergleich der URIs werden symbolische Links und Anomalien wie doppelte Slashes im Pfad normalisiert.

Für die Auflösung von URI-Konflikten stehen im Dialogfenster folgende Möglichkeiten zur Auswahl:

- **Vorschau:** Durch Klick auf den verlinkten Dokumententitel können Sie sich eine Vorschau des Dokumentes in einem neuen Browserfenster anzeigen lassen.
- **Status:** Ein Dokument befindet sich entweder "im Workflow" oder "im Archiv".
- **Bearbeiten:** Sie verwerfen das aktuelle Dokument und wählen stattdessen ein bestehendes zur Bearbeitung aus. Dabei wird das aktuelle Dokument komplett gelöscht und das ausgewählte durchläuft den Workflow. Sie führen dann den entsprechenden Workflow-Schritt des jeweiligen Dokuments aus. Bei Dokumenten, die aus dem Archiv importiert werden, ist das der auf den DocSelector-Schritt folgende Workflowschritt. Dokumente, die sich bereits im Workflow befinden, bearbeiten Sie an der entsprechenden Stelle weiter.
- **Kopieren:** Sie kopieren den Inhalt des ausgewählten Dokuments in das aktuelle und gehen weiter zum nächsten Workflow-Schritt. Wählen Sie diese Möglichkeit, um ein neues Dokument zu erstellen, dessen Inhalt auf dem einer älteren Version basiert.

- **Neues Dokument:** Sie verlassen den DocSelector-Schritt ohne weitere Aktion. Das aktuelle Dokument gelangt ohne Änderung zum nächsten Workflow-Schritt.

Die Konfiguration des Plug-Ins nehmen Sie im Workflow-Grid vor. Dafür stehen Ihnen folgende Parameter zur Verfügung:

Angezeigte Dokumenteninformationen:

Normalerweise wird in der Liste existierender Dokumente jeweils der Titel angezeigt bzw. "[ohne Titel]", wenn ein Dokument keinen Titel hat. Da alle Workflow-Parameter der Template-Expandierung unterworfen sind, entspricht die Anzeige des Titels einem Zugriff auf <!--XX-title-->. Wenn Sie beispielsweise den Eigentümer des Dokumentes hinter dem Titel in Klammern anzeigen lassen möchten, ändern Sie dies wie folgt um:

```
<!--XX-title--> (User: <!--USER_CONF:name-->)
```

Dialog bei einfacher Auswahl nicht anzeigen:

Mit dieser Option aktivieren Sie folgendes Verhalten:

1. Existiert bereits genau ein Dokument mit identischem URI und kann das Dokument nicht kopiert werden, wird die letzte Version des Dokuments in den Workflow importiert. Dabei wird kein interaktiver Dialog angezeigt und das aktuelle Dokument wird automatisch komplett gelöscht.

Dieser Fall tritt entweder ein, wenn Sie keine Berechtigung zur Erstellung von Kopien in diesem Workflow-Schritt haben oder die Erstellung von Kopien allgemein deaktiviert wurde.

Tatsächlich unterbinden Sie so die Entstehung von mehreren Dokumenten mit identischem URI in der betreffenden Rubrik. Wenn Sie ein neues Dokument erstellen, importieren Sie stattdessen automatisch die letzte Version aus dem Archiv und bearbeiten diese, wobei das neue Dokument automatisch gelöscht wird.



Hinweis:

Diese Variante eignet sich besonders, wenn Sie nur ein Dokument in einer Rubrik anlegen und dieses im Laufe der Zeit ändern möchten. Da immer die letzte Version automatisch aus dem Archiv importiert und überschrieben wird, sorgen Sie auch dafür, dass nur wenige archivierte Versionen des Dokumentes vorliegen.

Zugriffe auf die Menüpunkte "freigeschaltete Dokumente bearbeiten" oder das Archiv sind dabei nicht nötig. Auf diese Weise minimieren Sie den Schulungsaufwand für Ihre Redakteure und beschleunigen gleichzeitig die Arbeitsabläufe, da beispielsweise Zugriffe auf das Archiv je nach Umfang des Projektes sehr ressourcenintensiv sein können.

Wenn Sie garantieren möchten, dass in einer Rubrik nicht mehrere Dokumente den selben URI haben, wählen Sie außerdem die Optionen "Kopieren von Dokumenten deaktivieren" (siehe Abschnitt 8.5.10, „Kopieren von Dokumenten deaktivieren“ und Abschnitt 8.5.10, „Überschreiben von Dokumenten deaktivieren“). Gegenüber einer Einschränkung der Rollenberechtigungen hat dies den Vorteil, dass Benutzer mit Zugriff auf die Superuser-Rolle diese Regel nicht unbeabsichtigt umgehen können.

2. Sollte unter der URI nicht genau ein Dokument existieren, bekommen Sie den Dialog zur interaktiven Auflösung des URI-Konfliktes angezeigt.

Kopieren von Dokumenten deaktivieren:

Wenn Sie diese Option markieren, können Sie keine Kopie mit identischem URI auf der Basis eines bestehenden Dokumentes erzeugen. Sie setzen damit alle vorgegebenen Kopierberechtigungen für diesen Workflow-Schritt außer Kraft.

Überschreiben von Dokumenten deaktivieren:

Wenn Sie diese Option markieren, können Sie bereits bestehende Dokumente mit identischem URI bearbeiten oder mit einem neuen Dokument überschreiben. Dabei basiert der Inhalt des neuen Dokumentes auf dem des bereits bestehenden. Mit dieser Option setzen Sie alle etwaigen Berechtigungen zur Erzeugung neuer Dokumente außer Kraft. Diese Einstellung kann auch vom Superuser nicht umgangen werden.

Kopieren:

Wählen Sie hier die Rollen aus, die Kopien auf der Basis bestehender Dokumente erzeugen dürfen.



Wichtig:

Beachten Sie, dass Benutzer mit Zugriff auf die Superuser-Rolle keinerlei Beschränkungen durch diese Option unterliegen.

Dokument erstellen:

Mit dieser Option schränken Sie die Rollen ein, die "neue Dokumente" erzeugen dürfen.



Hinweis:

Tatsächlich ist ja bereits ein neues Dokument erzeugt worden, wenn Sie diesen Workflowschritt erreichen. "Dokument erstellen" bedeutet hier also genau genommen, dass das aktuelle Dokument nicht gelöscht wird.



Wichtig:

Beachten Sie, dass Benutzer mit Zugriff auf die Superuser-Rolle keinerlei Beschränkungen durch diese Option unterliegen.

8.5.11 Eraser

Setzen Sie das Eraser-Plug-In ein, um ein Dokument einschließlich aller eventuell bestehenden Kopien vom Zielsystem zu löschen. Das Dokument kann anschließend weiter im Workflow bearbeitet werden. Die Archiv-Version(en) des gelöschten Dokumentes bleiben erhalten. Das Plug-In benötigt keine weitere Konfiguration.

8.5.12 Exists

Mit dem Exists-Plug-In prüfen Sie, ob ein Dokument mit identischem URI bereits den Workflow durchlaufen hat. Das bedeutet, dass es ein anderes Dokument mit den selben Werten in den Metafeldern *directory* und *filename* gibt, dessen Status **beendet** ist.

Trifft diese Bedingung zu, wird das Plug-In durch den Südausgang ("Ja") verlassen, andernfalls durch den Ostausgang ("Nein").



Hinweis:

Beachten Sie, dass es sich dabei um ein **anderes** Dokument handeln muss. Ist das Dokument mit dem selben URI lediglich eine frühere Version des aktuell im Workflow befindlichen, wird dies so behandelt, als ob es noch kein Dokument mit diesem URI gäbe. Das ist notwendig, um Schwierigkeiten bei der Bearbeitung bereits freigeschalteter Dokumente zu vermeiden.

Dieses Plug-In benötigt keine Konfiguration.

Ältere Imperia-Versionen lösen etwaige URI-Konflikte beim Freischaltvorgang einfach durch Überschreiben des bestehenden Dokuments. Mit Hilfe des Exists-Plug-Ins haben Sie die Möglichkeit, Vorkehrungen für den Fall zu treffen, dass ein Dokument mit dem selben URI existiert. Typischerweise positionieren Sie das Plug-In dort im Workflow, wo der URI des Dokuments sich nicht mehr ändern wird, also "stabil" ist.

8.5.13 Expiry

Mit Hilfe dieses Plug-Ins wird das Ablaufdatum eines Dokuments bestimmt. Der User erhält beim Erreichen dieses Workflow-Schrittes eine Eingabemaske, in der er das Ablaufdatum über eine Pull-Down-Liste bestimmt. Damit das Dokument beim Erreichen des eingestellten Datums gelöscht wird, muss der Hintergrund-Daemon auf dem Produktions- und Zielsystem laufen.

Im Grid können Sie die Zugriffsberechtigungen für das Plug-In einstellen. Legen Sie fest, welche Rollen ein Dokument in diesem Schritt bearbeiten, lesen oder löschen dürfen.

8.5.14 Genehmigung

Dieses Plug-In stellt eine Prüfinstanz für Dokumente dar (Mehr-Augen-Prinzip). Beide Ausgänge des Genehmigung-Plug-Ins müssen mit einem anderen Plug-In verbunden werden, so dass Dokumente, abhängig davon, ob der User genehmigt oder verwirft, einen anderen Weg durch den Workflow nehmen.

Wird der südliche Ausgang nicht verbunden, gibt das Plug-In eine Warnung aus, da in einem solchen Fall ein abgelehntes Dokument in der Freischaltliste erscheinen kann. Der südliche Ausgang wird verwendet, wenn eine Genehmigung nicht erteilt wird, der östliche Ausgang wird verwendet, wenn die Genehmigung erteilt wird.

Das Plug-In interpretiert nicht, ob das mit dem östlichen Ausgang verbundene Plug-In nach einer Genehmigung sinnvoll ist. Es bietet eher die Möglichkeit, den Workflow interaktiv zu verzweigen.

Das Genehmigung-Plug-In kann im Workflow nicht umgangen werden. Über die Konfiguration des Plug-Ins steuern Sie, wer den Genehmigungsschritt ausführen darf.

Bearbeiten

Rollen mit dieser Berechtigung dürfen ein Dokument genehmigen bzw. ablehnen.

Lesen

Wählen Sie aus, welche Rollen Dokumente, die sich in diesem Schritt befinden, auf dem Schreibtisch sehen können. Allerdings können nur Benutzer mit dem Bearbeiten-Recht den Workflowschritt auch ausführen.

Löschen

Diese Berechtigung ermöglicht das Löschen von zu genehmigenden Dokumenten vom Schreibtisch.

Lesen Sie hierzu das Kapitel **Dokumente genehmigen** im Userhandbuch.

8.5.15 If-Else

Dieses Plug-In wird benutzt, um den Weg von Dokumenten in Abhängigkeit von einer oder mehreren Bedingungen zu bestimmen. Es besitzt drei Ausgänge und benötigt keine Usereingabe.

Trifft die erste Bedingung zu, wird der nördliche Ausgang des Plug-Ins verwendet. Trifft die erste Bedingung nicht zu, wird die zweite Bedingung überprüft.

Trifft die zweite Bedingung zu, wird der östliche Ausgang benutzt. Trifft die zweite Bedingung ebenfalls nicht zu, wird der südliche Ausgang benutzt.

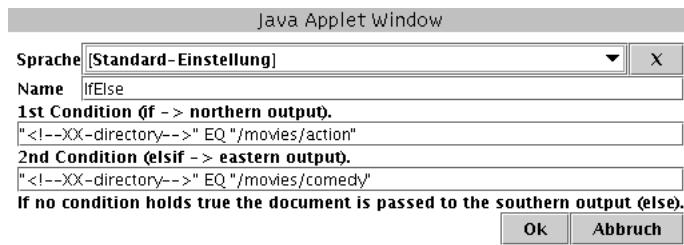


Abb. 8.10: Eigenschaften des IfElse-Plug-Ins

Die Bedingungen werden im Eigenschaften-Dialog des Plug-Ins eingegeben. Die Syntax hierfür ist weitgehend identisch mit der Syntax für IF-Abfragen im Template. Sie müssen lediglich die Klammern weglassen.

Bleibt eine der Eingabezeilen leer, gilt die damit verbundene Bedingung als nicht erfüllt.

Achten Sie bei der Eingabe der Bedingungen auf Leerzeichen, Tabstopps, Zeilenumbrüche und ähnliches. Leerzeichen am Anfang und am Ende einer Eingabezeile werden nicht beachtet. Mehrere Leerzeichen innerhalb des Bedingungstextes werden zu einem Leerzeichen zusammengefasst.

8.5.16 Genehmigung mit Link-Check

Dieses Plug-In stellt eine Kombination der Link-Check-Funktion, die über den Schreibtisch verfügbar ist, und dem Genehmigungs-Plug-In dar.

Führen Sie den Workflow-Schritt aus, um den Dialog **Link-Checker** zu öffnen:



Abb. 8.11: Dialog Link-Checker

Interne Links

Dieser Bereich zeigt das Ergebnis der Linküberprüfung. Die aufgeführten Dateien sind verlinkt, so dass Sie diese im Browser aufrufen können. Dateien, bei denen die Linkprüfung fehlgeschlagen ist, können Sie mit einer nebenstehenden Checkbox markieren und anschließend für die markierten Dateien eine **Überprüfung der markierten Links mit HTTP-Requests** anstoßen. Klicken Sie hierzu auf den entsprechend benannten Link in diesem Bereich.

Ergebnisse des Link-Checks (Linkcheck)

Die hier angezeigten Bedienelemente entsprechen denen des Genehmigungs-Plug-Ins. Lesen Sie hierzu auch das Kapitel **Dokumente genehmigen** im Userhandbuch.

Genehmigte Dokumente verlassen den Workflow-Schritt über den Ostausgang und gelangen zum nachfolgenden Schritt. Abgelehnte Dokumente verlassen diesen Schritt über den Südausgang.

Alternativ können Sie auch auf den Button **Schreibtisch** klicken, um zum Schreibtisch zurückzugelangen. Das Dokument bleibt dann im Link-Check. Eine Weiterleitung zum nächsten Workflow-Schritt ohne die Genehmigung des Dokuments ist nicht möglich.

Konfiguration des Plug-Ins

Im Grid können Sie festlegen, welche Rollen zur Ausführung dieses Schritts berechtigt sind.

8.5.17 Metaedit

Dieses Plug-In zeigt die Eingabemaske der Metadatei an und speichert die dort eingegebenen Daten.

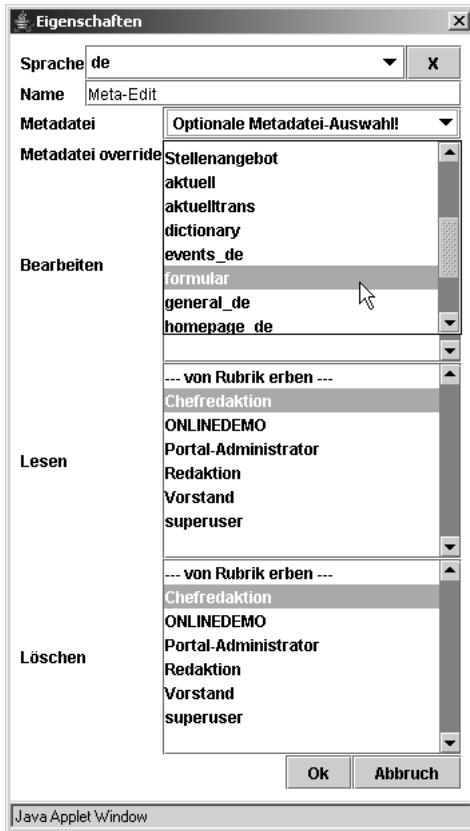


Abb. 8.12: Eigenschaften des Metaedit-Plug-Ins

Folgende Konfigurationsmöglichkeiten stehen zur Auswahl:

Metadatei

Über eine Auswahlliste können Sie für diesen Workflow-Schritt eine Metadatei auswählen, die von den allgemeinen Einstellungen für die Rubrik abweicht.

Metadatei override

Tragen Sie hier den Namen einer alternativen Metadatei ein, die in diesem Schritt zum Einsatz kommen soll. Sie können hier auch über XX-Variablen Ergänzungen vornehmen.

Bearbeiten

Über eine Auswahlliste können Sie festlegen, welche Rollen ein Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahlliste können Sie bestimmen, welche Rollen ein Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

8.5.18 ■ Metasetter

Dieses Plug-In erlaubt es Ihnen, innerhalb des Workflows beliebigen Metafeldern Werte zu zuweisen. Sowohl der Name als auch der Wert des Metafeldes werden wie in einem Template expandiert.

Beispiel:

```
last_change      <!--Xstrftime:en_US:%c-->
```

Hiermit wird der Wert der Meta-Variablen `last_change` auf Datum und Zeit nach der `en_US_Locale` gesetzt. Würde beispielsweise eine Meta-Variable `timestamp` o.ä. existieren, könnte man auch als Schlüssel `<!--XX-timestamp-->` verwenden.

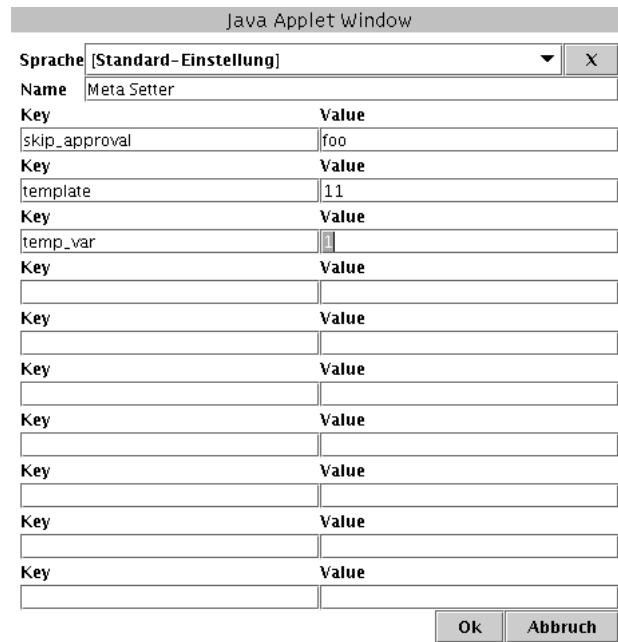


Abb. 8.13: Eigenschaften MetaSetter

Zurzeit können in einem MetaSetter-Plug-In gleichzeitig zehn Meta-Variablen gesetzt werden. Es ist jedoch nicht möglich, einer einzelnen Meta-Variablen mehrere Werte zu zuweisen.

8.5.19 MetaSplice

Das MetaSplice-Plug-In bildet die Funktionalität der Perl-Funktion `splice()` zur Bearbeitung von Metadaten ab. Die Konfiguration des Plug-Ins verhält sich analog zum Aufruf der Perlfunktion. Wie beim Perl-Splice können Sie mit diesem Plug-In je nach Konfiguration also die Funktionen `shift()`, `unshift()`, `pop()` und `push()` auf Daten in Metafeldern anwenden.

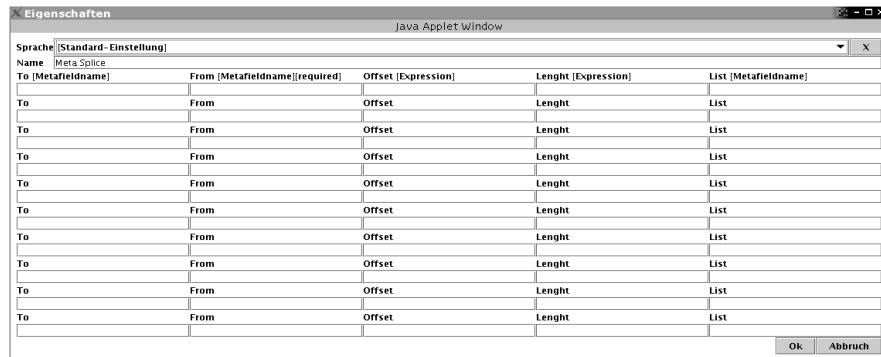


Abb. 8.14: Eigenschaften des Plug-Ins MetaSplice

Damit lassen sich zum Beispiel Workflows mit Schleifen umsetzen, wie das folgende Beispiel zeigt:

8.5.19.1 Mehrsprachige Workflows mit MetaSplice

Das MetaSplice-Plug-In ermöglicht die Umsetzung eines Workflows zur Erstellung mehrsprachiger Seiten. Dabei wird für jede gewünschte Sprache eine Copy-Seite erzeugt. Jede dieser Copy-Seiten befüllt der Benutzer durch einen mehrfach ausgeführten Bearbeiten-Schritt mit Inhalt. Sind alle Sprachversionen erstellt, verlässt das Dokument den Workflow.

Das könnte beispielsweise folgendermaßen aussehen:

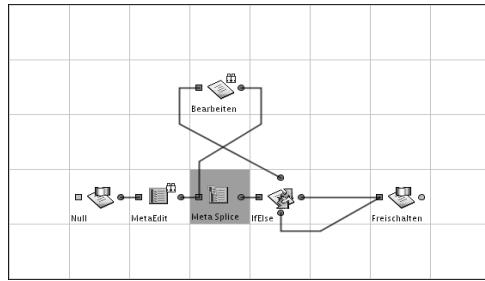


Abb. 8.15: mehrsprachiger Workflow mit MetaSplice

Entscheidend ist die Konfiguration des MetaSplice-Plug-Ins und des If-Else-Plug-Ins, wodurch der Bearbeiten-Schritt mehrfach ausgeführt werden kann. Die Konfiguration der beiden Plug-Ins sieht wie folgt aus:

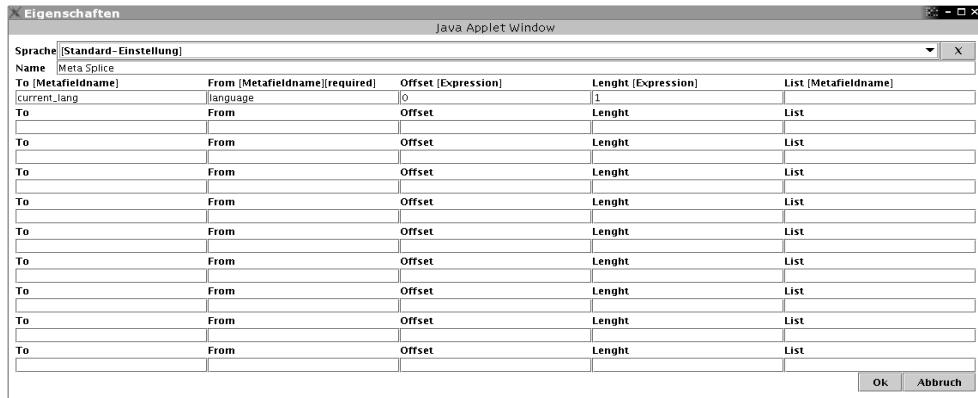


Abb. 8.16: Konfiguration von MetaSplice

Das MetaSplice-Plug-In entfernt hier bei jedem Aufruf des Workflow-Schrittes genau den ersten Wert (Spalten **Offset** und **Length**) aus einem Metafeld namens *language* (Spalte **From**) und speichert diesen in einem anderen Metafeld namens *current_lang* ab (Spalte **To**). Dies entspricht der Arbeitsweise der Perl-Funktion *shift()*.

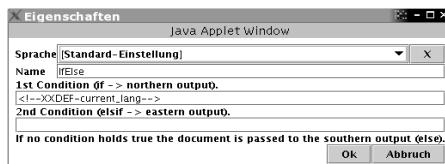


Abb. 8.17: Konfiguration von IfElse

Nach dem Bearbeiten-Schritt prüft das If-Else-Plug-In, ob das Feld *current_lang* einen Wert enthält. Trifft dies zu, verlässt das Dokument den Schritt über den Nord-Ausgang, der mit dem MetaSplice-Plug-In verbunden ist. Die beiden anderen Ausgänge führen zum Ende des Workflows, der in der obigen Abbildung zur besseren Übersichtlichkeit durch ein umbenanntes Null-Plug-In repräsentiert wird.

8.5.20 Mülleimer

Dieses Plug-In entfernt Dokumente aus dem Workflow ohne weitere Abfrage. Es entfernt jedoch keine archivierten Dokument-Versionen oder zu einem Dokument gehörende Dateien (Bilder, Multimedia-Objekte etc.).

8.5.21 Multilanguage

Mit dem Plug-In Multilanguage steuern Sie die Ersetzungen, die der Templateprozessor für die Verwaltung mehrsprachiger Inhalte vornimmt. Setzen Sie es im Workflow vor dem Edit-Plug-In ein.

8.5.22 Null

Bei diesem Plug-In handelt es sich um eine Beispiel-Plug-In-Implementierung ohne weitere Funktionalität. Es dient als Rumpf für die Entwicklung eigener Plug-Ins. Ferner kann es dazu verwendet werden, die Übersichtlichkeit eines stark verzweigten Workflows zu verbessern.

Das Null-Plug-In eignet sich auch als "Dummy-Workflow-Eingang", der Änderungen am Anfang des eigentlichen Workflows ermöglicht.

Zu jeder Rubrik, in der Dokumente erstellt werden können, gehört ein Workflow, den die Dokumente der Rubrik durchlaufen. In den Rubrikeinstellungen wird dabei auch immer der Eingang des Workflows gespeichert, beispielsweise "Meta-Edit". Wenn Sie jedoch einen Workflow verändern, indem Sie dem ursprünglichen Eingang ein anderes Plug-In voranstellen, beispielsweise den DocSelector, kann die Referenz auf den ursprünglichen Workflow-Eingang nicht mehr aufgelöst werden. Dadurch produzieren alle Rubriken, die den veränderten Workflow nutzen, plötzlich eine Fehlermeldung und es lassen sich darin keine Dokumente mehr anlegen.

Stellen Sie hingegen Ihrem Workflow ein Null-Plug-In voran, ist es vollkommen unproblematisch, zwischen Null-Plug-In und eigentlichem Workflow-Beginn weitere Workflow-Schritte einzuhängen. In den Rubriken, die den betreffenden Workflow nutzen, ist das Null-Plug-In als Eingang gespeichert worden und dies ist auch nach der Änderung der erste Schritt.

8.5.23 Publish

Dieses Plug-In gibt dem Benutzer die Möglichkeit, ein Veröffentlichungsdatum einzugeben. Das Dokument wird automatisch freigeschaltet, wenn dieses Datum erreicht wird. Dieses Plug-In benötigt einen gestarteten Hintergrund-Daemon auf dem Produktionssystem.

8.5.24 Shredder

Dieses Plug-In löscht Dokumente aus dem Workflow ohne weitere Sicherheitsabfrage. Gleichzeitig werden auch alle archivierten Versionen dieses Dokuments und alle eingebetteten Objekte entfernt.

8.5.25 SmartMeta

Dieses Plug-In verändert Metadaten eines Dokumentes anhand einer vorher definierten Mapping-Tabelle. Beispielsweise können Usereingaben in Schlagwörter für Suchmaschinen umgewandelt werden, ohne dass eine manuelle Eingabe des Users notwendig ist.

8.5.26 Unter-Workflow

Mit diesem Plug-In können Sie komplexe Workflows in Module unterteilen. Diese Module funktionieren ähnlich wie Codeincludes und können in mehreren Haupt-Workflows eingesetzt werden.

Der Inhalt eines Unter-Workflows wird beim Start des Hauptworkflows eingelesen, so dass der Ablauf genau so ist, als ob die einzelnen Schritte direkt im Hauptworkflow eingebunden wären.

In den Plug-In-Einstellungen wählen Sie aus, welcher Workflow geladen werden soll.

Dies geschieht über die Option *Eingang des Unter-Workflows*.

Dazu ein Beispiel:

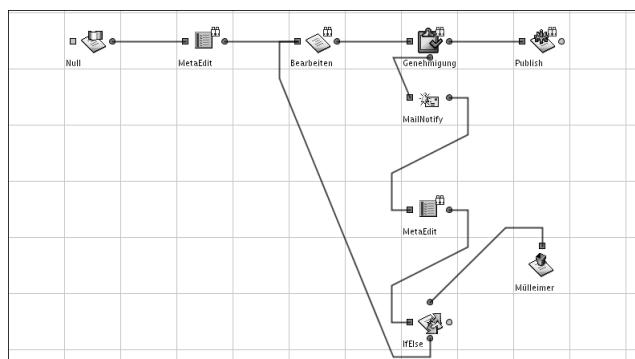


Abb. 8.18: Beispiel-Workflow ohne Unter-Workflows

Im oben stehenden Workflow muss ein Dokument vor seiner Veröffentlichung z.B. vom Chefredakteur genehmigt werden.

Falls das Dokument nicht genehmigt wird, kann der Chefredakteur noch zwischen zwei Möglichkeiten wählen. Entweder muss das Dokument korrigiert oder komplett neu erstellt werden. Im ersten Fall wird es zurück in den Bearbeitungs-Schritt geleitet, im zweiten gelöscht. Außerdem wird der zuständige Redakteur per Email benachrichtigt.

Der komplette Zweig zur Behandlung nicht genehmigter Dokumente soll nun in einen Unter-Workflow ausgegliedert werden.

Dazu wird zunächst ein eigener Workflow angelegt, der die Schritte aus diesem Zweig enthält. Die folgende Abbildung zeigt den Aufbau dieses Workflows:

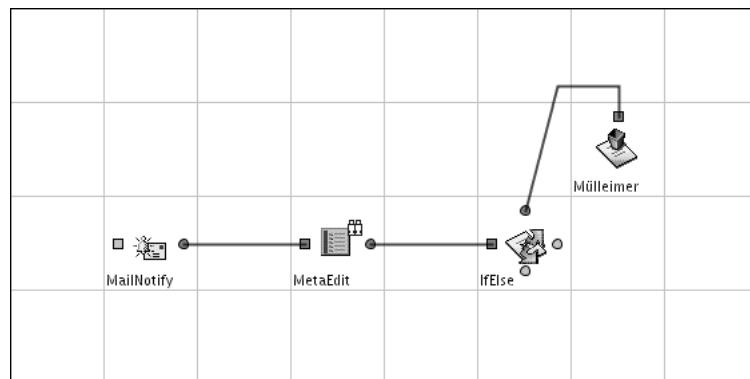


Abb. 8.19: Der Unter-Workflow

Im Haupt-Workflow wird der entsprechende Zweig entfernt und durch ein Unter-Workflow-Plug-In ersetzt:

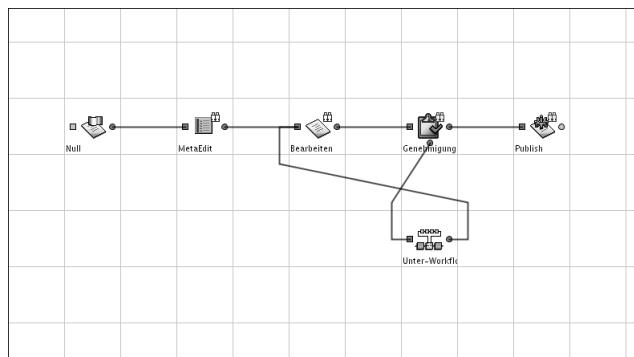


Abb. 8.20: Der abgeänderte Haupt-Workflow

In der Plug-In-Konfiguration wird festgelegt, welcher Workflow in den Haupt-Workflow geladen werden soll.



Abb. 8.21: Konfiguration des Unter-Workflows

8.5.27 Upload

Dieses Plug-In kommt nur im Zusammenhang mit Asset-Dokumenten zum Einsatz. Es steuert die Speicherung der mit einem Asset-Template erfassten Metainformationen sowie die Übertragung der Asset-Datei von Ihrem lokalen zum Imperia-Entwicklungssystem. Es zählt zu den semi-interaktiven Plug-Ins, da Sie beim Mehrfachupload nicht für jedes Asset einen eigenen Workflow-Schritt ausführen müssen, sondern alle Assets in einem Durchgang hochladen. Folgende Konfigurationsmöglichkeiten stehen Ihnen zur Verfügung:

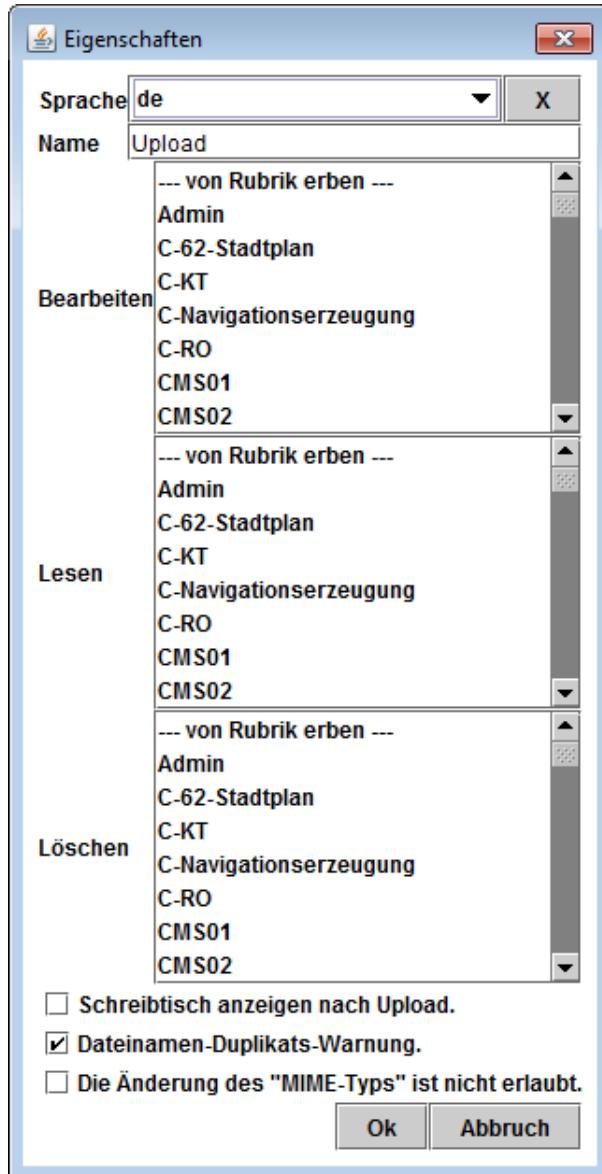


Abb. 8.22: Konfiguration des Upload-Plug-Ins

Bearbeiten

Über eine Auswahlliste können Sie festlegen, welche Rollen ein Dokument bearbeiten dürfen, das sich in diesem Workflow-Schritt befindet.

Lesen

Über eine Auswahlliste können Sie bestimmen, welche Rollen ein Dokument auf dem Schreibtisch sehen, das sich in diesem Workflow-Schritt befindet.

Löschen

Wählen Sie aus, welche Rollen ein Dokument vom Schreibtisch löschen dürfen, das sich in diesem Workflow-Schritt befindet.

Schreibtisch anzeigen nach Upload

Normalerweise springt der Workflow nach dem Upload-Schritt automatisch weiter zum nachfolgenden Workflow-Schritt. Aktivieren Sie diese Option, wenn Sie möchten, dass stattdessen nach erfolgtem Upload der Schreibtisch erscheint.

Dateinamen-Duplikatswarnung

Wenn Sie diese Option aktivieren, prüft Imperia vor dem Speichern eines Assets im Upload-Schritt, ob in der betreffenden Rubrik bereits ein Asset mit diesem Dateinamen vorliegt. Ist dies der Fall bleibt der Workflow im Upload-Schritt und eine Warnmeldung erscheint:

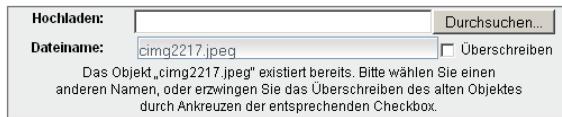


Abb. 8.23: Warnmeldung bei Dateinamensgleichheit

Sie können dann entweder durch Markieren der Option **Überschreiben** das bestehende Asset ersetzen lassen oder einen anderen Dateinamen vergeben.



Hinweis:

Beim Mehrfachupload greift diese Option nicht. Sollten Sie nicht im voraus die Option Überschreiben aktiviert haben, wird dort ein Asset bei Dateinamensgleichheit grundsätzlich nicht hochgeladen.

Die Änderung des "MIME-Typs" ist nicht erlaubt

Diese Option greift ein, wenn ein bestehendes Asset erneut importiert werden soll. Sie verhindert dabei, dass ein Objekt mit einem anderen Dateiformat als das bestehende hochgeladen wird.

Wenn das Asset zum Beispiel ein JPEG-Bild war, wird mit dem Setzen des Häkchens verhindert, dass ein PNG-Bild hochgeladen wird. Dies dient dazu, dass keine Bilddaten unter einer falschen Endung (hier: PNG-Bild mit der Endung .jpg) in das Asset Management eingespielt werden.

8.5.28 Zufall

Dieses Plug-In besitzt drei Ausgänge, von denen es einen zufällig auswählt und das Dokument an das daran anschließende Plug-In weitergibt.

Jedem der Ausgänge kann eine Gewichtung zugewiesen werden, durch die die Verteilung der Dokumente auf die einzelnen Ausgänge beeinflusst werden kann. Zur Berechnung der Gewichtung für einen Ausgang wird die Summe der drei zugewiesenen Gewichtungen durch den dem betreffenden Ausgang zugewiesenen Wert geteilt.

Wird beispielsweise dem nördlichen Ausgang eine Gewichtung von 4, dem östlichen eine Gewichtung von 10 und dem südlichen eine Gewichtung von 6 zugewiesen, werden die Dokumente wie folgt verteilt:

- 20% nach Norden
- 50% nach Osten
- 30% nach Süden

Anstelle von Zahlen können die Parameter auch vom Template-Prozessor berechnet werden, beispielsweise `<!--xx-time_to_spare-->`. Sollte ein Parameter leer oder 0 sein, werden diese Parameter behandelt, als hätten sie den Wert 1, um eine Division durch 0 zu vermeiden.

Dieses Plug-In scheint auf den ersten Blick ein reines Spielzeug zu sein, wenn Sie jedoch einige hundert Dokumente von einem externen Content-Provider in Imperia importieren, die bearbeitet oder genehmigt werden müssen, kann mit Hilfe dieses Plug-Ins die Last automatisch auf mehrere User verteilt werden.

Kapitel 9. Menü-Management

In diesem Kapitel finden Sie Informationen über die Verwaltung der Imperia-Menüs und der Zugriffsberechtigungen der Rollen auf Menüs und Untermenüs. Für jede im System eingerichtete Rolle kann eine eigene Kombination von Menüpunkten konfiguriert werden, die dann für alle User, die diese Rolle annehmen, gilt.

Es können auch eigene Untermenüs angelegt und mit Icons versehen werden. Zu einem Menü gehören folgende Teile:

- ein Symbol,
- Zugriffsrechte für Rollen,
- ein Link zum Aufrufen des entsprechenden Skripts,
- ein Linktext, der zum Öffnen des Menüs dient und
- eine Beschreibung.

Alle Arbeiten an den Menüs werden im Dialog **Menü bearbeiten** durchgeführt. Dieser Dialog ist in drei Abschnitte gegliedert:

- Menüabkürzungen,
- Zusätzliche Links,
- Menü-Konfiguration.

Menüabkürzungen

In diesem Abschnitt werden die ID und der Titel des Menüs festgelegt. Wenn Sie das Hauptmenü bearbeiten, können Sie die ID nicht ändern. Beim Erstellen oder Bearbeiten von Untermenüs können Sie sowohl die ID als auch den Titel ändern.



Hinweis:

Bei Änderungen an der ID eines bestehenden Untermenüs erzeugt Imperia automatisch ein neues Untermenü mit der neuen ID.

In diesem Bereich sehen Sie außerdem eine Liste aller bereits vergebenen IDs sowie weiterer ebenfalls reservierter Begriffe, die Sie nicht als IDs verwenden dürfen.

Zusätzliche Links

Dieser Abschnitt gibt Ihnen die Möglichkeit, in dem Menü, das Sie bearbeiten, zusätzliche Buttons anzeigen zu lassen, mit denen die User aus dem Menü heraus bestimmte Funktionen aufrufen können.

Der Abschnitt ist als Tabelle aufgebaut, die aus den Spalten **Symbol**, **Rolle**, **Link-Text**, **Beschreibung** und **Link** besteht.

9.1 Untermenü hinzufügen

In neuen Untermenüs können zum Beispiel die Imperia-Funktionen zusammengefasst werden, die nur einer bestimmten Rolle zugewiesen wurden. Jeder User sieht in jedem Menü nur die Menüpunkte, die seiner aktuellen Rolle zugeordnet sind.

Um ein neues Menü zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü hinzufügen**.
Es öffnet sich der Dialog **Menü bearbeiten**.
2. Tragen Sie im Abschnitt **Menüabkürzungen** eine ID und einen Titel für das neue Menü ein.

Unterhalb der Eingabefelder werden alle IDs aufgelistet, die schon vergeben sind. Diese können Sie nicht verwenden.

3. Nun müssen Sie die einzelnen Menüpunkte erstellen bzw. konfigurieren. Dazu steht der Bereich **Menü-Konfiguration** zur Verfügung.

Geben Sie zunächst in der ersten Tabellenzeile in der Spalte **Symbol** die Nummer des Symbols an, das für den ersten Menüpunkt verwendet werden soll. In Imperia stehen ca. 80 Symbole zur Auswahl. Diese sind über eine Select-Box in der Spalte **ICON** auswählbar. Sobald Sie eines der Icons aus der Liste ausgewählt haben, bekommen Sie oberhalb der Select-Box eine Vorschau angezeigt. Vorher sehen Sie dort ein Fragezeichen.

Wählen Sie dann in der Spalte **Rollen** eine oder mehrere Rollen aus, die auf den Menüpunkt zugreifen dürfen. Markieren Sie dazu die entsprechenden Rollen in der Liste. Mehrere Rollen werden mit der Maus und gedrückter Strg-Taste markiert.

Tragen Sie in der Spalte **Link** den Link ein, den der Menüpunkt ausführen soll. Zum Beispiel <http://www.imperia.de> oder /cgi-bin/site_general_settings.pl.

Geben Sie in der Spalte **Linktext** den Text ein. Dieser Text erscheint später fett neben dem Symbol. Der User kann wahlweise durch Anklicken des Symbols oder des Textes den Link ausführen. Tragen Sie in der Spalte **Beschreibung** einen kurzen Text ein, der dem User die Funktion des Menüpunkts verdeutlichen soll.

Führen Sie diese Schritte für jeden neuen Menüpunkt aus und verwenden Sie für jeden Menüpunkt eine eigene Tabellenzeile.

4. Klicken Sie auf den Button **Okay**, um das neue Menü zu speichern. Der Button **Abbrechen** verlässt den Dialog, ohne Eingaben zu speichern.

Neue Menüdateien werden im Verzeichnis /site/config abgelegt. Der Dateiname wird aus der Menü-ID und der Endung .mnu gebildet.

Das Erstellen eines neuen Menüs ist damit abgeschlossen. Das neue Menü kann nun der Ordnerliste hinzugefügt werden. Lesen Sie hierzu das folgende Kapitel.

9.1.1 Untermenü einem Ordner hinzufügen

Jedes Imperia-Menü enthält drei nicht belegte Menüpunkte, die Sie verwenden können, um eigene Einträge zu machen. Sollte diese Anzahl nicht ausreichen, speichern Sie zunächst das Menü. Nach dem Speichern stehen erneut drei freie Menüpunkte zur Verfügung.

Um einen neuen Menüpunkt in die Linkliste des Hauptmenüs aufzunehmen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** den Menüpunkt **Hauptmenü bearbeiten**.
- Es öffnet sich der Dialog **Menü bearbeiten**.
2. Tragen Sie in der Spalte **Symbol** in einer freien Zeile die Symbolnummer ein, die verwendet werden soll.
3. Markieren Sie in der Spalte **Rolle** die Rollen, die zum Zugriff auf das neue Menü berechtigen.
4. Geben Sie nun in der Spalte **Link** folgenden Link ein:

/cgi-bin/site_main.pl

5. Ergänzen Sie diesen Link durch den Aufruf des neuen Menüs:

?Menüname

Wenn Sie ein Menü mit der ID *menue1* erzeugt haben, muss der Link wie folgt aussehen:

/cgi-bin/site_main.pl?menuel

6. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Änderungen zu speichern.
Sie gelangen in einen Dialog, von dem aus Sie das geänderte Hauptmenü über einen Testlink anzeigen lassen können.
7. Mit dem Button **Okay** gelangen Sie zum Menü-Manager, den Sie über den Button **Schließen** verlassen können. Sie gelangen dann automatisch in das Hauptmenü.

9.2 Menüs bearbeiten

Mit Hilfe dieser Funktion werden der Aufbau der Menüs und die Zugriffsrechte auf die Menüpunkte einge stellt.

Sie können die im Hauptmenü angezeigten Texte verändern, weitere Menüpunkte dem Hauptmenü hinzufügen, vorhandene Menüpunkte aus dem Hauptmenü entfernen und die Rollen auswählen, die zum Zugriff auf die einzelnen Menüpunkte berechtigen.

9.2.1 Menüpunkt zum Hauptmenü hinzufügen

Um einen Menüpunkt hinzuzufügen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Hauptmenü bearbeiten**.
Es öffnet sich der Dialog **Menü bearbeiten**.
2. Ändern Sie in den einzelnen Abschnitten die Angaben und Einstellungen.
Näheres hierzu erfahren Sie in den folgenden Kapiteln.
3. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Änderungen und Einstellungen zu speichern.
Der Button **Abbrechen** schließt den Dialog, ohne Änderungen zu speichern.

9.2.2 Menüpunkt einem Untermenü hinzufügen

Um einen Menüpunkt einem Untermenü hinzuzufügen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü bearbeiten**.
Es öffnet sich eine Liste aller eingerichteten Untermenüs.
2. Wählen Sie das Untermenü, dem Sie einen weiteren Menüpunkt hinzufügen wollen, aus.
Es öffnet sich der Dialog **Menü bearbeiten** mit den Daten des gewählten Menüs.
3. Tragen Sie in einer freien Zeile die erforderlichen Daten des neuen Menüpunktes ein. Verfahren Sie hier wie in Abschnitt 9.1 **Untermenü hinzufügen** auf Seite 128 beschrieben.
4. Klicken Sie auf den Button **Okay** am unteren Rand des Dialogs, um Ihre Änderungen zu speichern.
Über den Button **Abbrechen** verlassen Sie den Dialog, ohne Änderungen zu speichern.
Es erscheint ein Hinweis, dass das Menü gespeichert wurde, den Sie ebenfalls mit dem Button **Okay** bestätigen müssen. Sie gelangen daraufhin in den Menü-Manager.
5. Schließen Sie den Menü-Manager mit Hilfe des Buttons **Schließen**. Sie gelangen daraufhin in das geänderte Hauptmenü.

9.2.3 Zugriffsrechte für Menüs einstellen

Um die Zugriffsrechte für Menüs einzustellen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü bearbeiten**.
Es öffnet sich eine Liste aller eingerichteten Untermenüs.
2. Wählen Sie das Untermenü, dem Sie einen weiteren Button hinzufügen wollen, aus.
Es öffnet sich der Dialog **Menü bearbeiten**, mit den Daten des gewählten Menüs.
3. Markieren Sie nun für jeden Menüpunkt in der Spalte **Rolle** die Rollen, die zum Zugriff berechtigen bzw. heben Sie die vorhandene Markierung für nicht zum Zugriff berechtigte Rollen auf.
Um eine Markierung aufzuheben, klicken Sie bei gedrückter Strg-Taste auf den markierten Eintrag. Mehrere Markierungen werden mit Hilfe der Strg-Taste getätigt.

9.2.4 Untermenü löschen

Um ein Untermenü zu löschen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Untermenü löschen**.
Es wird eine Liste aller Untermenüs angezeigt.
2. Klicken Sie auf das Menü, das gelöscht werden soll.
Es erscheint eine Sicherheitsabfrage, die Sie mit **OK** bestätigen müssen. Daraufhin wird das Untermenü gelöscht und eine Bestätigung wird angezeigt.
3. Klicken Sie auf den Button **Okay**, um in das Hauptmenü zurückzugelangen.



Hinweis:

Die Standardmenüs von Imperia lassen sich nicht löschen. Dies ist nur bei nachträglich erstellten Menüs möglich.

9.3 Menüstruktur anzeigen

Um die vorhandene Menüstruktur anzeigen zu lassen, gehen Sie wie folgt vor:

1. Klicken Sie im Ordner **Menüs** auf den Menüpunkt **Menüstruktur anzeigen**.
Es wird eine Liste aller Menüs und Untermenüs sowie der dazugehörigen Links angezeigt.
2. Klicken Sie auf den Button **Zurück** am unteren Rand der Liste, um diese Ansicht wieder zu verlassen.

Kapitel 10. Media-Asset-Management (MAM)

Das Media-Asset-Management, kurz MAM, dient der Verwaltung unstrukturierter Daten und ermöglicht prozessorientiertes Dokumenten-Management.



Hinweis:

Aus Kompatibilitätsgründen ist die bisherige Mediendatenbank zunächst weiterhin in Imperia enthalten. Um diese anstatt des Media-Asset-Managements zu verwenden, setzen Sie die in der Datei site/config/system.conf die Variable MAM_DISABLE auf den Wert 1. Hinweise zum Gebrauch der bisherigen Mediendatenbank entnehmen Sie bitte der Dokumentation zu Imperia 8.

Media-Assets sind Imperia-Dokumente

Unstrukturierter Content wie Grafiken, Sound- und Video-Dateien sowie PDF- und Office-Dokumente lassen sich als Imperia-Dokumente im System verwalten. Die unstrukturierten Dokumente werden so zum integralen Bestandteil des prozessorientierten Content-Managements. Der Upload erfolgt Workflow-basiert. Hierfür stehen Workflow-Plug-Ins zur Verfügung. Weitere Plug-Ins zur Manipulation der Metadaten lassen sich hinzufügen. Da Assets Imperia-Dokumente sind, versioniert und archiviert sie das System automatisch. So haben Sie jederzeit Zugriff auf ältere Versionen eines Assets und auf die zugehörige Änderungshistorie.

Media-Assets sind frei strukturierbar

Im Media-Asset-Management sind Sie hinsichtlich der Strukturierung Ihrer Medienobjekte sehr flexibel. Sie können beispielsweise die Rubrikenstruktur Ihres Contents im Media-Asset-Management ohne Redundanzen abbilden. Bei den Verzeichnissen im MAM handelt es sich um Imperia-Rubriken, über deren Einstellungen Sie die Zugriffsberechtigungen für die darin befindlichen Assets festlegen.

Media-Assets und Linkmanagement

Im Media-Asset-Management können Sie nachvollziehen, ob Referenzen auf ein Asset bestehen. Referenzierte Assets lassen sich nur nach einer zusätzlichen Bestätigung löschen.

10.1 Mit Asset-Rubriken arbeiten

Eine Asset-Rubrik besitzt alle Eigenschaften einer herkömmlichen Dokumenten-Rubrik in Imperia. Sie können in den Einstellungen einer Asset-Rubrik Metainformationen hinterlegen, die für alle darin angelegten Assets verfügbar sind. In einer Rubrik erzeugte Assets folgen einem definierten Workflow und erhalten per Default den Dateinamen, den Sie in den Rubrikeinstellungen festlegen. Für den Upload von Assets und das Anlegen von Asset-Rubriken können Sie Templates definieren. Lesen Sie Abschnitt 10.1.1 **Asset-Rubrik anlegen** auf Seite 132 für weitere Informationen zu den Eigenschaften von Asset-Rubriken. Einzelne Mediendateien finden Sie auch im Archiv über die zugehörige Rubrik.

Für jede Asset-Rubrik gibt es auch ein Verzeichnis im Dateisystem, in dem die Dokumente, sprich Assets, dieser Rubrik abgelegt sind. Das ist aber lediglich eine von vielen Eigenschaften der Mediendateien und nicht mehr ausschlaggebend für deren Strukturierung und Verwaltung.

10.1.1 Asset-Rubrik anlegen

Markieren Sie zunächst im Verzeichnisbaum des Media-Asset-Managements die Rubrik, in der Sie eine neue Unterrubrik anlegen möchten. Danach rufen Sie die Funktion **Datei->Rubrik erzeugen** auf.

Um eine neue Rubrik für Assets zu erstellen, benötigen Sie eine Rolle, die in der Ausgangsrubrik unter "Rollen, die diese Rubrik verändern dürfen" eingetragen ist (siehe Abschnitt 6.1 **Rubriken-Eigenschaften bestimmen** auf Seite 85). Ist dies nicht der Fall, erscheinen die Menüpunkte ausgegraut.

Wenn Sie kein eigenes Template für das Anlegen von Rubriken erstellt haben, gelangen Sie in den Dialog "**Neue Rubrik**". Ab hier ähnelt der Ablauf dem des Anlegens neuer Dokument-Rubriken (siehe Abschnitt 4.1.1 **Neue Rubrik anlegen** auf Seite 55 und Abschnitt 6.1 **Rubriken-Eigenschaften bestimmen** auf Seite 85). Legen Sie folgende Einstellungen für die neue Asset-Rubrik fest:

Rubriken-Informationen, Workflow und Template

Name

Der Name der Rubrik. Unter diesem Namen finden Sie die Rubrik im Media-Asset-Management, in der Rubriken-Verwaltung und im Archiv.

Beschreibung

Tragen Sie in diesem Feld einen optionalen Beschreibungstext für die Rubrik ein. Dieser erscheint im Archiv als Tooltip für den Info-Button einer Rubrik.

Verzeichnis

Legen Sie das Verzeichnis bzw. den Pfad fest, in dem die Assets in dieser Rubrik auf dem Server liegen.

Dateiname

Hier geben Sie den Default-Dateinamen für in der Rubrik erstellte Dokumente an. Für den Upload von Assets ist dieser Eintrag unerheblich, da Sie den Dateinamen des Assets im Verlauf des Upload-Workflows eintragen.

Nur Unterrubriken

Wählen Sie diese Option, wenn Sie in der Rubrik keine Assets, sondern nur Unterrubriken anlegen möchten.

Dokumente und Unterrubriken

Mit dieser Option können Sie in der Rubrik sowohl Unterrubriken als auch Assets anlegen.

Assetspeicherung erlauben?

Diese Option ermöglicht das Anlegen von Assets in der betreffenden Rubrik. Sie müssen Sie nur bei der Hauptrubrik des Media-Asset-Managements markieren. Bei allen darunter liegenden Asset-Rubriken ist sie automatisch aktiviert.

Workflow-Auswahl

Wählen Sie hier den Workflow **MAM**, der den Ablauf des Uploads von Assets regelt.

Metadatei

Da der Workflow für den Asset-Upload keinen Meta-Edit-Schritt beinhaltet, ist keine Metadatei notwendig. Tragen Sie in dieses Feld **dummy** ein.

Template

Wählen Sie hier das Template für den Asset-Upload aus. Sie können hierfür das Standard-Template verwenden oder ein eigenes Template erstellen. Lesen Sie hierzu den Abschnitt "**Datei-Upload (Media-Asset-Management)**" in der Syntax-Referenz für Imperia-Templates im Programmierhandbuch.

Zugriffsrechte

In diesem Bereich bestimmen Sie, welche Rollen in welcher Form auf die Rubrik und die darin befindlichen Assets zugreifen dürfen.



Hinweis:

Die hier vorgenommenen Einstellungen sind äquivalent zu den globalen und den verzeichnisbezogenen Berechtigungen für die Mediendatenbank aus älteren Imperia-Versionen.

Rollen, die diese Rubrik lesen dürfen

Rollen, die diese Berechtigung haben, sehen den Inhalt der betreffenden Asset-Rubrik. Sie ermöglicht die Verwendung der darin befindlichen Assets in Dokumenten. Außerdem muss ein Benutzer auch dann Leserecht für eine Rubrik haben, wenn er Zugriff auf deren Unterrubriken haben soll.

Rollen, die Dokumente in dieser Rubrik anlegen dürfen

Diese Berechtigung ermöglicht das Anlegen von Assets in dieser Rubrik.

Rollen, die Dokumente in dieser Rubrik löschen dürfen

Mit dieser Berechtigung ist das Löschen von Assets und Unterrubriken in der betreffenden Rubrik erlaubt.

Rollen, die diese Rubrik verändern dürfen

Benutzer mit Rollen, die hier eingetragen sind, dürfen Unterrubriken erstellen und die Einstellungen der Rubrik ändern.

Speichern Sie die neu angelegte Rubrik durch Klick auf **Okay**. Sie gelangen daraufhin ins Rubriken-Management von Imperia, wo Sie die soeben erstellte Rubrik sehen können.

10.1.2 Asset-Rubrik bearbeiten

Wählen Sie zunächst im Rubrikenbaum des Media-Asset-Managements die gewünschte Rubrik aus. Rufen Sie anschließend die Funktion **Datei->Rubrik bearbeiten** auf. Sie gelangen zum Dialog "**Rubrik bearbeiten**". Die Einstellungsmöglichkeiten ähneln weitgehend denen des Dialogs "**Neue Rubrik**" (siehe vorhergehender Abschnitt und Abschnitt 6.1 **Rubriken-Eigenschaften bestimmen** auf Seite 85). Zusätzlich haben Sie noch die Möglichkeit, beliebige Metainformationen für die Rubrik zu erfassen. Weitere Einzelheiten zu Rubriken-Metainformationen finden Sie im Abschnitt 4.1.2 **Rubriken bearbeiten** auf Seite 56. Nehmen Sie die gewünschten Einstellungen vor und speichern Sie diese durch Klick auf **Okay**.

10.1.3 Asset-Rubrik löschen

Zum Löschen einer Asset-Rubrik wählen Sie zunächst die gewünschte Rubrik im Rubrikenbaum des Media-Asset-Managements aus. Danach rufen Sie die Funktion **Datei->Rubrik löschen** auf. Sie können nur leere Rubriken löschen. Befinden sich noch Assets und / oder Unterrubriken in einer Rubrik, erhalten Sie bei einem Löschversuch die Fehlermeldung "Rubrik ist nicht leer". Der Löschvorgang wird in diesem Fall abgebrochen. Entfernen Sie zunächst alle Assets bzw. Unterrubriken aus der Rubrik bzw. verschieben Sie diese in eine andere Asset-Rubrik und führen Sie dann den Löschvorgang erneut aus.

10.1.4 Einzelne Assets und Asset-Rubriken verschieben

Diese Funktion steht nur Benutzern mit Zugriff auf die Superuser-Rolle zur Verfügung. Sie können wahlweise ganze Asset-Rubriken oder einzelne Assets verschieben. Rufen Sie hierzu die Funktion **Hauptmenü->Struktur->Assets verschieben** auf. Daraufhin öffnet sich folgender Dialog:

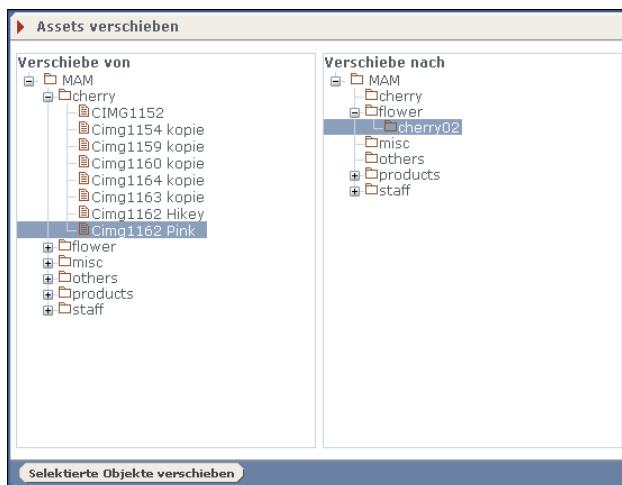


Abb. 10.1: Der Dialog Assets verschieben mit ausgeklappter Rubrikenstruktur

Dabei ist die Rubrikenstruktur in beiden Dialoghälften bei jedem Aufruf des Dialogs eingeklappt. Klappen Sie also zunächst die Rubrikenstruktur entsprechend aus, so dass die zu verschiebende Rubrik bzw. das zu verschiebende Asset sowie die Zielrubrik, in die Sie verschieben möchten, sichtbar sind.

Markieren Sie unter **Verschiebe von** das Asset bzw. die Rubrik, die Sie verschieben möchten, und unter **Verschiebe nach** die gewünschte Zielrubrik. Anschließend führen Sie mit einem Klick auf **Selektierte Objekte verschieben** die Verschiebe-Aktion aus.

Nachdem Sie die Aktion ausgeführt haben, lädt sich das Dialogfenster erneut mit vollständig eingeklappten Rubrikbäumen.



Wichtig:

Der Hintergrund-Daemon Hermes (siehe Kapitel 11 Der Daemon Hermes) muss auf Develop und Live gestartet sein, damit Verschiebe-Aktionen wirksam werden.

Neben der Änderung der Metainformationen verschiebt der Hermes-Daemon auch die Asset-Dateien einschließlich aller zugehörigen Varianten auf dem Develop- und dem Live-System. Etwaige auf verschobene Assets referenzierende Dokumente schaltet der Daemon mit aktualisierten Pfadangaben erneut frei, ohne dass die betroffenen Dokumente den Workflow durchlaufen. Es fallen dabei also keine zusätzlichen Archivversionen der betroffenen Dokumente an.

10.2 Import von MDB-Inhalten in das Media-Asset-Management

Sollten Sie eine bestehende Imperia-Installation mit Inhalten auf die Version 8.5 aktualisieren, ist eine Migration der Inhalte der MDB notwendig, um diese im Media-Asset-Management verfügbar zu machen. Hierfür nutzen Sie das Skript `site/bin/mam_import_mdb.pl`.

Das Skript kennt folgende Parameter:

Parameter	Beschreibung
<code>-l, --list</code>	Ausgabe von Informationen über die möglichen und benötigten Parameter und gültige Werte für diese.
<code>-t, --template=TEMPLATE</code>	Template für die Rubriken des Media-Asset-Managements. Geben Sie den Namen des Templates ohne Dateiendung an.
<code>-w, --workflow=WORKFLOW</code>	Workflow für die Rubriken des Media-Asset-Managements. Wenn Sie keinen anderen Workflow definiert haben, geben Sie den mitgelieferten Workflow "MAM" an. Notieren Sie den Parameter in der Form <i>ID:WORKFLOW-Eingang</i> .
<code>-r, --root=NODE_ID</code>	NodeID der Wurzelrubrik für Media Assets. Wenn Sie diesen Parameter verwenden, muss die betreffende Rubrik bereits bestehen.
<code>-h, --help</code>	Ausgabe einer kurzen Hilfseite.
<code>-v, --verbose</code>	Ausführliche Ausgabe.

Tabelle 10.1. Parameter für `mam_import_mdb.pl`

10.2.1 Durchführung des Imports

Beim Import der Mediendatenbank-Inhalte gehen Sie folgendermaßen vor.

Hauptrubrik für das Media-Asset-Management erstellen

Für den Import der MDB-Inhalte in das Media-Asset-Management benötigen Sie eine Hauptrubrik, in der die Option **Assetspeicherung erlauben?** aktiviert ist. Die notwendigen Schritte zum Erstellen dieser Rubrik sind in Abschnitt 10.1.1 **Asset-Rubrik anlegen** auf Seite 132 beschrieben.

Objekte importieren

Rufen Sie das Skript `site/bin/mam_import_mdb.pl` mit den Parametern auf, die im vorhergehenden Abschnitt beschrieben sind.

Beispiel für einen Aufruf:

```
perl mam_import_mdb.pl -t upload_01 -w MAM:1 -r 321 -v
```

Das Skript importiert die Struktur Ihrer Mediendatenbank ins Asset-Management. Dabei erzeugt es für jede Medientyp-Kategorie der MDB eine eigene Unterrubrik im MAM. Nach erfolgtem Import befinden sich also unter der von Ihnen spezifizierten Hauptrubrik des Media-Asset-Managements die Rubriken **IMAGES**, **VIDEO**, **AUDIO**, **CONTENT**. In diesen Rubriken finden Sie dann die jeweiligen Inhalte aus der ursprünglichen Mediendatenbank.

Zur Übernahme bestehender Referenzen aus der MDB müssen Sie anschließend noch einmalig das Skript `site/bin/admin_rebuild_references.pl` aufrufen:

```
perl site/bin/admin_rebuild_references.pl --new --mdb
```



Warnung:

*Verwenden Sie den Parameter `--mdb` nur, nachdem Sie die Migration der MDB in das Media-Asset-Management ausgeführt haben, und auch nur **einmalig!** Wenn das Skript mit diesem Parameter einmal durchgelaufen ist, erzeugt die nochmalige Verwendung des Parameters fehlerhafte Referenzen!*

Für weitere Hinweise zur Verwendung rufen Sie das Skript mit dem Parameter `--help` auf.

10.3 Import von Assets aus einem Ordner oder einer Zip-Datei

Möchten Sie eine größere Anzahl von Assets in das Media-Asset-Management importieren, so können Sie dies entweder auf der Kommandozeile oder über eine grafische Oberfläche erledigen.

Das Kommandozeilentool ist mächtiger; kleinere Anzahlen von Assets ohne besondere Anforderungen können bequemer über die Oberfläche importiert werden.

10.3.1 Vorarbeiten und Konfiguration

Bevor Sie jedoch erfolgreich importieren können, müssen einige Einstellungen gemacht werden.

Setzen Sie die System.conf-Variable `MAM_GET_UPLOAD_FILENAME` auf den Wert 1. Siehe Abschnitt 13.4.20.6 Variable `MAM_GET_UPLOAD_FILENAME` auf Seite 220.

Standardmäßig werden Zip-Dateien aus dem Verzeichnis `/spool` importiert. Sie können über die System.conf-Variable `MAM_SPOOL_PATH` aber ein anderes Verzeichnis, das unterhalb des site-Verzeichnisses liegen muss, wählen. Legen Sie zunächst das Verzeichnis, aus dem die Dateien importiert werden sollen, an und sorgen Sie für Leserechte des Webservers in diesem Verzeichnis. Verzeichnisrechte können für das Spool-Verzeichnis und die direkten Unterverzeichnisse über „Spool Chmod“ eingerichtet werden.

Beispiel:

```
"MAM_SPOOL_PATH" = "/mam_import"
```

Mit der Variable `MAM_MAX_SPOOL_SIZE` können Sie die Maximalgröße für den Import von Assets festlegen. Der erwartete Wert der Variablen ist die Byte-Anzahl. Ist der Wert nicht vorhanden, wird keine Größenbeschränkung vorgenommen.

Beispiel, Beschränkung auf 100 MB:

```
"MAM_MAX_SPOOL_SIZE" = "104857600"
```

Durch Einfügen der Variablen `PLUGIN_ARCHIVE_ZIP_BIN` in die Datei `system.conf` können Sie festlegen, welche Bibliothek für das Entpacken verwendet werden soll. Ist die Variable nicht vorhanden oder der Wert leer, wird versucht, das PERL-Modul `Archive::Zip` zu verwenden, falls das installiert ist.

Beispiel:

```
"PLUGIN_ARCHIVE_ZIP_BIN" = "/usr/bin/unzip"
```

**Hinweis:**

Beachten Sie, dass ZIP-Dateien bestimmte Dateigrößen nicht überschreiten sollten, sonst kommt es beim Entpacken zu Problemen. Bei 32-Bit-Systemen darf die Größe 2 GB nicht überschreiten. Auf 64-Bit-Systemen können ZIP-Dateien bis 4 GB entpackt werden.

10.3.2 Import mit Hilfe des Kommandozeilentools

Das Kommandozeilenskript zum Import wird folgendermaßen aufgerufen: Wechseln Sie in das Basisverzeichnis Ihrer Imperia-Installation des Entwicklungsservers. Rufen Sie `site/bin/mam_spool_import.pl` mit den passenden Parametern auf. Folgende Parameter sind möglich:

- -f, --file=Dateiname

Dateiname, relativ zum Spool-Verzeichnis (üblicherweise `site/spool`, kann aber in der `system.conf` geändert werden).

Als Dateiname kann eine einzelne Datei, ein Verzeichnis, das Dateien enthält, oder ein Zip-Archiv angegeben werden. Es können mehrere Optionen -f gesetzt werden. Die Angabe mindestens eines Files ist Voraussetzung für den Programmablauf.

- -c, --catid=CATEGORY_ID

ID der MAM-Rubrik, in die importiert werden soll. Die Angabe einer Kategorie ist Voraussetzung für den Programmablauf.

- -u, --uid=UID

Imperia-User-ID des Mitarbeiters, der als Ersteller der MAM-Objekte fungieren soll.

- -e, --extract

Dateien aus einem Zip-File sollen entpackt werden.

- -l, --follow-cat

MAM-Kategorien sollen für Verzeichnisse in dem angegebenen Archiv oder Verzeichnis angelegt werden.

- -d, --bydir

Bei der Zuordnung von MAM-Kategorien zu importierten Kategorien soll der Verzeichnisname der MAM-Kategorie und nicht, wie voreingestellt, der Kategoriename verglichen werden.

- -w, --overwrite

Assets mit dem gleichen Pfad wie ein importiertes Asset sollen überschrieben werden.

- -b, --background

Das Skript soll im Hintergrund laufen und Ausgaben in das Logfile geschrieben werden.

- -t, --batch

Ausgaben des Skriptes sollen statt auf der Konsole in das Logfile geschrieben werden.

- -q, --quit

Tritt ein Fehler auf, soll der Programmablauf sofort gestoppt werden.

- -h, --help

Ein Hilfetext wird ausgegeben und der Import nicht durchgeführt.

- -v, --verbose

Ausführlichere Meldungen sollen ausgegeben werden.

Beispiele:

```
mam_spool_import.pl -c /1/15/143 -u 1 -f "company.jpg" -f "agb.zip" -f "logo.png"
```

Die Dateien company.jpg, agb.zip und logo.png werden mit dem Benutzer mit der ID 1 in die MAM-Rubrik /115/143 hochgeladen. Das Zip-Archiv wird nicht entpackt.

```
mam_spool_import.pl -c /12 -u 1 -e -l -f "images.zip"
```

Das Archiv images.zip wird entpackt und die enthaltenen Dateien werden in die MAM-Rubrik /12 bzw. deren Unterrubriken hochgeladen. Dabei werden die in dem Zip-Archiv enthaltenen Verzeichnisse mit den Rubrikennamen des MAM verglichen und die Assets in die vorhandenen Rubriken eingesortiert. Nicht bestehende Rubriken werden angelegt.

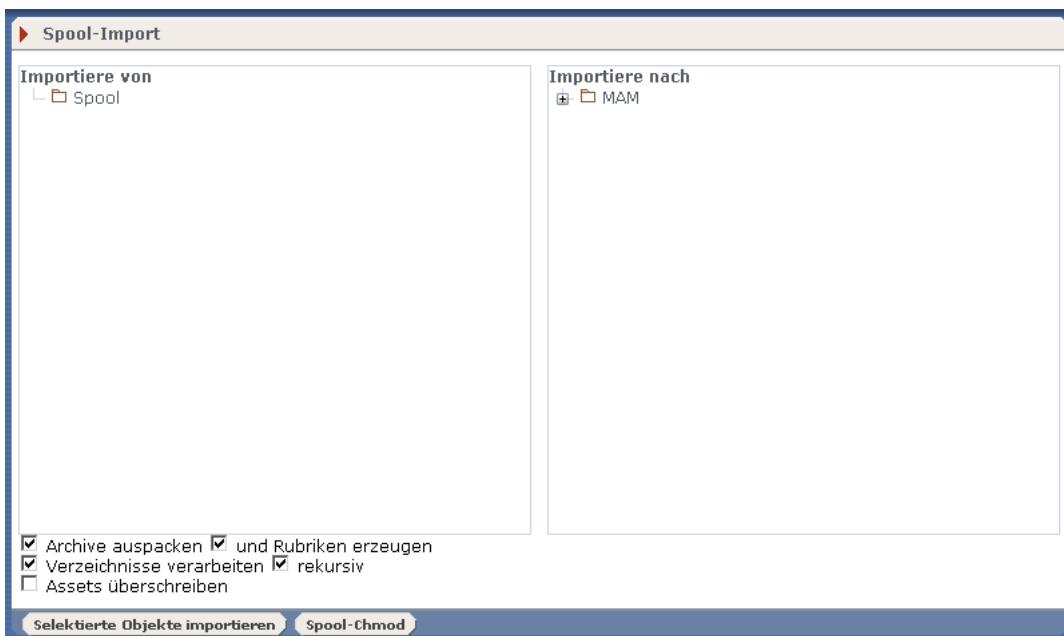
```
mam_spool_import.pl -c /12 -u 1 -l -d -w -f "importverzeichnis"
```

Die in dem Verzeichnis site/spool/importverzeichnis enthaltenen Dateien werden in die MAM-Rubrik /12 bzw. deren Unterrubriken hochgeladen. Dabei werden die in dem Ordner enthaltenen Verzeichnisse mit den Rubriken-Verzeichnisnamen verglichen und die Assets in die vorhandenen Rubriken eingesortiert. Nicht bestehende Rubriken werden angelegt. Vorhandene Assets werden gegebenenfalls überschrieben.

10.3.3 Import über die grafische Oberfläche

Die grafische Oberfläche des MAM-Imports rufen Sie wie folgt auf:

/cgi-bin/site_mam_spool_import.pl.



10.3.3.1 Vorarbeiten und Konfiguration

Bevor Sie jedoch erfolgreich importieren können, müssen einige Einstellungen gemacht werden.

Setzen Sie die System.conf-Variable MAM_GET_UPLOAD_FILENAME auf den Wert 1.

```
"MAM_GET_UPLOAD_FILENAME" = "1"
```

Standardmäßig werden Zip-Dateien aus dem Verzeichnis `/spool` importiert. Sie können über die System.conf-Variable `MAM_SPOOL_PATH` aber ein anderes Verzeichnis, das unterhalb des `site`-Verzeichnisses liegen muss, wählen. Legen Sie zunächst das Verzeichnis, aus dem die Dateien importiert werden sollen, an und sorgen Sie für Leserechte des Webservers in diesem Verzeichnis. Verzeichnisrechte können für das Spool-Verzeichnis und die direkten Unterverzeichnisse über "Spool Chmod" eingerichtet werden.

Beispiel:

```
"MAM_SPOOL_PATH" = "/mam_import"
```

Mit der Variable `MAM_MAX_SPOOL_SIZE` können Sie die Maximalgröße für den Import von Assets festlegen. Der erwartete Wert der Variablen ist die Byte-Anzahl. Ist der Wert nicht vorhanden, wird keine Größenbeschränkung vorgenommen.

Beispiel, Beschränkung auf 100 MB:

```
"MAM_MAX_SPOOL_SIZE" = "104857600"
```

Durch Einfügen der Variablen `PLUGIN_ARCHIVE_ZIP_BIN` in die Datei `system.conf` können Sie festlegen, welche Bibliothek für das Entpacken verwendet werden soll. Ist die Variable nicht vorhanden oder der Wert leer, wird versucht, das PERL-Modul `Archive::Zip` zu verwenden, falls das installiert ist.

Beispiel:

```
"PLUGIN_ARCHIVE_ZIP_BIN" = "/usr/bin/unzip"
```



Hinweis:

Beachten Sie, dass ZIP-Dateien bestimmte Dateigrößen nicht überschreiten sollten, sonst kommt es beim Entpacken zu Problemen. Bei 32-Bit-Systemen darf die Größe 2 GB nicht überschreiten; auf 64-Bit-Systemen können ZIP-Dateien bis 4 GB entpackt werden.

10.3.2 Die Web-Oberfläche

Assets aus dem Ordner `site/spool/test` werden in die MAM-Rubrik "Bilder Einsätze" importiert.

The screenshot shows the 'Spool-Import' dialog. In the 'Importiere von' section, 'Spool' is expanded, and 'test' is selected. In the 'Importiere nach' section, the site structure is shown: MAM, VIDEO, AUDIO, IMAGES, CONTENT, CMS02, CMS03, CMS04, Stadtbib, Feuerwehr, Bilder Einsatz (which is selected), Testverzeichnis, temporaer, Bilder Website, Musikakademie, Gesundheitsamt, PEG, Kulturprogramm, Sonderprojekte, Umweltportal, Langes Feld, Langes Feld-Klimagutachten, CMS01, AnmeldungunaMusikakademie. At the bottom left, there are checkboxes for 'Archive auspacken' (checked), 'Verzeichnisse verarbeiten' (checked), and 'Assets überschreiben' (unchecked). At the bottom right, there are two buttons: 'Selektierte Objekte importieren' and 'Spool-Chmod'.

"Archive auspacken" - importierte Archivdateien werden entpackt (derzeit nur Zip-Archive) und die enthaltenen Dateien importiert.

- "und Rubriken erzeugen" - beim Import werden MAM-Rubriken für jeden importierten Ordner erzeugt.
- "Verzeichnisse verarbeiten" - der Inhalt ganzer Verzeichnisse wird importiert.
- "rekursiv" - es werden auch alle Unterordner der gewählten Ordner verarbeitet.
- "Assets überschreiben" - Assets, die bereits bestehen, werden durch neu importierte mit dem gleichen Dateinamen überschrieben.

Spool Chmod

Für Redakteure, die die Superuser-Rolle haben, wird ein Button "Spool Chmod" angezeigt. Hierüber können Rechte für Importverzeichnisse gesetzt werden. Für jedes Directory (linker Teil des Fensters) können Zugriffsrechte für Rollen (rechts) gesetzt werden. Dies funktioniert in zwei Ebenen, also dem Spoolverzeichnis und einer Verzeichnisebene darunter.

10.4 Konfiguration des Media-Asset-Managements

Abgesehen von den benutzerabhängig gespeicherten Einstellungen des MAM, die im Benutzerhandbuch beschrieben sind, gibt es einige allgemeingültige Konfigurationsmöglichkeiten für das Media-Asset-Management. Diese Einstellungen nehmen Sie über Konfigurationsvariablen vor, die Sie in der Datei site/config/system.conf notieren.

10.4.1 Globale Suchfilter definieren

Wie im Abschnitt "Im Media-Asset-Management suchen" des Benutzererhandbuchs beschrieben, lassen sich Suchfilter abspeichern. Diese gelten allerdings nur für jeden Benutzer individuell. Als Benutzer mit Superuser-Zugang haben Sie jedoch auch die Möglichkeit, definierte Suchfilter so abzuspeichern, dass alle Benutzer Ihres Systems sie anwenden können. Markieren Sie hierfür beim Erstellen oder Bearbeiten eines Filters die Option "Filter global speichern" ("Save as global filter"). Das System speichert so erstellte Filter in der Variablen `MAM_SAVED_FILTERS`.

10.4.2 Asset-Metafelder global verfügbar machen

Das Media-Asset-Management erlaubt die Speicherung beliebiger Metainformationen zu den Assets. Um diese Metainformationen in Dokumenten abrufen zu können, müssen Sie die Metafelder global verfügbar machen. Hierzu verwenden Sie folgende Variable:

```
"GLOBAL_METAKEYS" = "Metafeldname01, Metafeldname02, MetafeldnameNN"
```

10.4.3 Asset-Metafelder in der Listenansicht anzeigen lassen

Globale Asset-Metafelder lassen sich auch als Spalten in der Listenansicht sowie im "Details"-Dialog eines Assets des Media-Asset-Managements anzeigen. Dies geschieht mit der Variablen `MAM_ADD_USER_FIELDS`. Syntax:

```
"MAM_ADD_USER_FIELDS" = "Metafeldname01:Spaltenüberschrift01, MetafeldnameNN  
: SpaltenüberschriftNN"
```



Hinweis:

Obenstehender Code ist aus Darstellungsgründen umgebrochen.

10.4.4 Titel-Leiste des MAM-Rubrikenbaums festlegen

Per Default ist der Rubrikenbaum im Media-Asset-Management mit **Navigation** betitelt. Diesen Titel ändern Sie mit der folgenden Variablen:

```
"MAM_NAV_HEADER" = "Asset-Rubriken"
```

10.4.5 Sortierung des Asset-Rubrikenbaums festlegen

Die Sortierung der Rubriken im Rubrikenbaum des Media-Asset-Managements erfolgt per Default in der Reihenfolge der Erstellung der Rubriken. Sie können dies auf alphabetische Sortierung umstellen, indem Sie die folgende Variable zur Systemkonfiguration setzen:

```
"MAM_CAT_ORDER_BY_NAME" = "1"
```

10.4.6 Template für die Erstellung von Asset-Rubriken festlegen

Wenn Sie für die Erstellung von Asset-Rubriken ein eigenes Template nutzen, geben Sie dessen Namen mit der Variablen `MAM_CATEGORY_TEMPLATE` an. Notieren Sie den Namen ohne Pfadangabe oder Dateiendung:

```
"MAM_CATEGORY_TEMPLATE" = "Templatename"
```

Templates für die Erstellung von Asset-Rubriken speichern Sie im Verzeichnis `site/templates` ab. Die Syntax für Asset-Rubrik-Templates entspricht der anderer Imperia-Templates. Ein Beispieltemplate finden Sie unter `site/config/templatecatmeta.htm.sample`.

10.4.7 Standard-Anzahl für Mehrfachupload definieren

Die Option **Datei->Mehrfachupload** erlaubt das Anlegen mehrerer Assets in einem Arbeitsgang. Mit der Variablen `MAM_MULTIUPLOAD_SIZE` legen Sie die Standardanzahl hierfür fest.

Beispiel:

```
"MAM_MULTIUPLOAD_SIZE" = "5"
```

10.4.8 Mehrfachupload-Optionen einrichten

Mit der Variablen `MAM_MULTIUPLOAD_OPTIONS` richten Sie die verfügbaren Optionen für den Mehrfachupload ein. Die Angabe der Optionen erfolgt in Form einer kommaseparierten Liste von Zahlen.

Beispiel:

```
"MAM_MULTIUPLOAD_OPTIONS" = "2,3,4,5,6,7,8,9,10"
```



Hinweis:

Ab einer Anzahl von ca. 15 Assets ist mit steigenden spürbaren Performance-Einbußen zu rechnen. Das ist natürlich nur ein Richtwert. Je nach System-Gegebenheiten kann es dazu auch schon früher kommen.

10.4.9 Linkanzeige der Listenansicht konfigurieren

Mit der Variablen `MAM_LINKS_STYLE` legen Sie die Darstellung der Links in den Spalten der Listen-Ansicht fest.

Die Syntax lautet wie folgt:

```
"MAM_LINKS_STYLE" = "[Option]"
```

Diese drei Optionen stehen zur Auswahl:

ICON

In den Spalten erscheinen lediglich kleine Icon-Grafiken.

TEXT

In den Spalten erscheinen ausschließlich Textlinks.

ICON, TEXT

In den Spalten erscheinen sowohl Icons als auch Text. Diese Einstellung ist der Default.

10.4.10 Asset-Details fixieren

Im Dialog Asset-Details können Benutzer die Metainformationen eines Assets ändern oder Felder hinzufügen und löschen. Die Felder `directory` und `filename` sind per Default fixiert, um die Datenintegrität von Assets und Dokumenten, die darauf verweisen, zu gewährleisten. Um weitere Felder zu fixieren, verwenden Sie die Variable `MAM_READ_ONLY_FIELDS`. Geben Sie die Felder, die nicht editierbar sein sollen, als kommasseparierte Liste an. Dazu ein Beispiel:

```
"MAM_READ_ONLY_FIELDS" = "height,width"
```

Obenstehendes Beispiel fixiert die Felder `height` und `width`. Sie können alle Felder aus dem Dialog "Asset Details" fixieren. Groß- und Kleinschreibung ist dabei irrelevant. Allerdings dürfen Sie bei mehreren Feldnamen keine Leerzeichen vor oder nach den Kommas setzen.

10.4.11 Felder verbergen

Wie im Abschnitt "Benutzeroberfläche des Media-Asset-Managements einrichten" des Benutzerhandbuchs beschrieben, lassen sich die Spalten der Listen-Ansicht ein- und ausblenden. Darüber hinaus können Sie Spalten der Listen-Ansicht sowie etwaige äquivalente Felder im Kontextmenü der Iconbox-Ansicht generell verbergen. Hierfür nutzen Sie die Variable `MAM_HIDDEN_FIELDS` mit folgender Syntax:

```
"MAM_HIDDEN_FIELDS" = "Regex, SpaltenID"
```

Tragen Sie die Felder, die Sie verbergen wollen, als kommasseparierte Liste mit oder ohne Leerzeichen als Wert ein. Die Felder können Sie mit der Feld-ID angeben. Alternativ kann die Angabe auch ohne das Präfix `_mam_` erfolgen. Die folgende Tabelle listet die verfügbaren Felder und zugehörigen IDs auf:

Deutscher Spaltenname (Listen-Ansicht)	FeldID
- (Checkbox)	<code>_mam_select</code>
Zoom	<code>_mam_zoom</code>
Icon	<code>_mam_icon</code>
Titel	<code>_mam_title</code>
Dateiname	<code>_mam_filename</code>
Verzeichnis	-
Dateigröße	<code>_mam_filesize</code>
Details	<code>_mam_details</code>
Referenzen	<code>_mam_references</code>
Löschen	<code>_mam_delete</code>
Erzeugt von	<code>_mam_createdby</code>
Erzeugungsdatum	<code>_mam_createdat</code>
Template	-
Abmessungen	<code>_mam_dimensions</code>
Benutzeraktion	<code>_mam_action</code>
Benutzervarianten	<code>_mam_variants</code>
Archiv	<code>_mam_archive</code>
Import	<code>_mam_reimport</code>

Tabelle 10.2. Verfügbare Felder im Media-Asset-Management



Hinweis:

Das Feld **Icon** lässt sich nur in der Listen-Ansicht verbergen.

10.4.12 Datenhaltung für Binärdateien bestimmen

Damit Assets versionierbar sind, legt Imperia eine Kopie der Assetdatei an und archiviert diese. Entsprechend der in den Systemeinstellungen gewählten Datenhaltung speichert Imperia diese Binärdaten im Dateisystem oder als BLOB-Felder in einer Datenbank. Wenn Sie eine SQL-Datenhaltung verwenden und nicht wünschen, dass Imperia die binären Assetdaten in der Datenbank speichert, setzen Sie in der Datei `site/config/system.conf` die Variable `BINARY_DOCTREE_PLUGIN` auf den Wert `File65`.

10.4.13 Datei-Benennung für Assets konfigurieren

Per Default legt Imperia neu erzeugte Assets unter dem Dateinamen im System ab, den Sie in den Einstellungen der betreffenden Asset-Rubrik festgelegt haben. Wie bei allen Imperia-Rubriken bzw. -Dokumenten ist der Standardwert hierfür "index.html". Asset-Dokumente stellen allerdings einen Sonderfall dar, da Imperia hier die Dateiendung automatisch an den jeweiligen MIME-Type der Asset-Datei anpasst. So wird beispielsweise eine PDF-Datei nicht als "index.html" abgespeichert, sondern als "index.pdf".

Sollte dieses Default-Verfahren zur Dateibenennung nicht Ihren Anforderungen entsprechen, können Sie es mit zwei Variablen beeinflussen.



Hinweis:

Neben den hier beschriebenen Möglichkeiten zur Beeinflussung des Default-Dateinamens von Assets können Sie diesen natürlich auch durch ein entsprechendes Eingabefeld im Upload-Template oder durch Workflow-Plug-Ins ändern. Einzelheiten hierzu entnehmen Sie bitte der Syntaxreferenz im Kapitel **Templates** des Programmierhandbuchs.

MAM_GET_UPLOAD_FILENAME

Setzen Sie diese Variable auf 1, um dem Benutzer beim Upload ein Eingabefeld für den gewünschten Dateinamen zur Verfügung zu stellen, wie es im folgenden Screenshot zu sehen ist:

The screenshot shows a user interface for file uploads. At the top, there is a button labeled 'Hochladen:' and a standard file selection input field. To its right is a 'Durchsuchen...' (Search...) button. Below this, there is another input field labeled 'Dateiname:' containing the value 'index'. To the right of this input field is a small checkbox labeled 'Überschreiben?' (Overwrite?).

Abb. 10.2: Upload-Feld mit Möglichkeit zur Eingabe des Dateinamens

Das Eingabefeld ist mit dem Standard-Dateinamen aus den Rubrikeinstellungen vorbelegt. Tragen Sie den gewünschten Dateinamen ein. Alternativ können Sie nur die Vorbelegung aus dem Eingabefeld entfernen, wenn Sie den ursprünglichen Dateinamen auch für das Asset automatisch übernehmen möchten.

Das Eingabefeld für den Dateinamen ist nur beim ersten Workflow-Durchlauf des Assets sichtbar. Bei einem Reimport erscheint es nicht mehr. Außerdem hat es Vorrang vor einem etwa von Ihnen im Upload-Template definierten Formularfeld zur Bestimmung des Dateinamens.

MAM_UNIQUE

Setzen Sie diese Variable auf 1, um sicherzustellen, dass neu erstellte Assets nicht aufgrund zufälliger Namensgleichheit Varianten anderer Assets überschreiben. Imperia speichert dann alle Assets inklusive der jeweiligen Varianten in einem eigenen Unterverzeichnis der betreffenden Asset-Rubrik.



Hinweis:

Wenn Sie dieses Verhalten nicht global über eine Systemkonfigurationsvariable aktivieren möchten, können Sie alternativ auch die Variable `_imperia_mam_unique` verwenden. Diese können Sie in den Rubrik-Metainformationen setzen oder auch beispielsweise mit einem Metasetter im Workflow. Beachten Sie dabei, dass Sie die Variable bereits vor dem Upload-Schritt setzen müssen, damit sie wirksam ist.

DISABLE_8BIT_FILERAMES

Derzeit unterstützt Imperia nicht die Verwendung von 8-bit-Zeichen in Dateinamen. Aus diesem Grund werden Umlaute, Leerzeichen, Sternchen, das Dollarzeichen usw. in Dateinamen in Unterstriche umgewandelt.

Bisher war es aber möglich, über den MAM-Upload Assets mit derartigen Datei- oder Verzeichnisnamen hochzuladen. Dies führte dann beim Einfügen in HTML-Seiten zu defekten Links. Der Grund für diese Freiheit war, dass Imperia nicht sicher herausfinden kann, in welchem Format die Zeichen im Betriebssystem kodiert sind.

Setzen Sie die system.conf-Variable `DISABLE_8BIT_Filenames` auf den Wert "1" um die Umschreibung von 8-bit-Datei- und Verzeichnisnamen in einen „sicheren“ Zeichensatz beim MAM-Upload zu erzwingen.

```
"DISABLE_8BIT_Filenames" = "1"
```



Hinweis:

Verwenden Sie diese Option nicht, wenn Ihr Dateisystem-Zeichensatz UTF-8 ist, es sei denn, Sie wollen wirklich Latin-1-Dateinamen erzwingen.

Um auch beim Massenimport von Assets die korrekte Übernahme von Zeichen wie deutschen Umlauten oder kyrillischen Zeichen in die Imperia-Oberfläche zu erreichen, muss auch die Variable `MAM_SPOOL_CHARSET` korrekt gesetzt sein.

MAM_SPOOL_CHARSET

Bei der Verwendung des MAM-Massenimports muss der Betriebssystem-Zeichensatz angegeben werden, um eine korrekte Übernahme von Sonderzeichen in die Imperia-Oberfläche und Logfiles zu erzielen.

Ist das Betriebssystem des Imperia-Entwicklungsservers Windows, können Sie als Wert "auto" eintragen. Imperia wird dann den Zeichensatz selbst ermitteln. Voreinstellung für `MAM_SPOOL_CHARSET` ist UTF-8. Bei Angabe eines Leerstrings wird "windows-1252" angenommen. Ist Ihnen bekannt, dass Ihr Betriebssystem-Zeichensatz UTF-8 ist, ist keine Umrechnung nötig. Kommentieren Sie dann `MAM_SPOOL_CHARSET` aus oder löschen Sie die Zeile aus der system.conf.



Hinweis:

Beim Import von Zip-Archiven ist die Umschreibung der Datei- und Verzeichnisnamen nicht möglich, da eine sichere Erkennung des von dem Zip-Tool beim Packen des Archivs verwendeten Zeichensatzes nicht möglich ist. Verwenden Sie daher, wenn möglich, direkt beim Packen des zu importierenden Archives die Zeichenkodierung UTF-8.

10.4.14 Unabhängige Freischaltung von Assets aktivieren

Normalerweise überträgt Imperia Assets erst dann auf das Zielsystem, wenn andere Dokumente auf sie referenzieren. Wenn Sie Assets unabhängig freischalten möchten, setzen Sie in der Datei `site/config/system.conf` die folgende Variable:

```
"MAM_ENABLE_PUBLISH" = "1"
```

Assets lassen sich dann automatisch und manuell aus der Freischaltliste heraus freischalten. Sie erscheinen außerdem in der Liste der freigeschalteten Dokumente.

10.4.15 Deaktivierung des Media-Asset-Managements

Wenn Sie statt des Media-Asset-Managements lieber weiterhin die Mediendatenbank verwenden möchten, können Sie dies mit der folgenden Konfigurationsvariablen erreichen:

```
"MAM_DISABLE" = "1"
```

Notieren Sie diese Variable in der Datei `site/config/system.conf`. Damit Sie die Mediendatenbank wieder über das Menü aufrufen können, müssen Sie einen entsprechenden Menüpunkt anlegen (siehe Kapitel 9 **Menü-Management**). Tragen Sie dabei in das Feld `Link` folgenden Wert ein:

MEDIADBWIN:/cgi-bin/site_md.pl



Wichtig:

Mit Imperia-Version 8.5 löst das Media-Asset-Management die Mediendatenbank ab. Diese wird jedoch aus Kompatibilitätsgründen vorübergehend weiter unterstützt.

10.4.16 Verfügbare Asset-Aktionen nachträglich festlegen

Diese Funktion steht nur in der Listen-Ansicht zur Verfügung. Wenn Sie nachträglich ändern möchten, welche Aktionen mit einem Asset möglich sein sollen, klicken Sie bei dem betreffenden Asset in der Spalte **Aktionen ändern** auf den gleichlautenden Link. Sie gelangen in den Dialog "Asset Aktionen", wo Sie durch Anklicken Aktionen aktivieren bzw. deaktivieren können. Klicken Sie auf "Speichern", um Ihre veränderten Einstellungen abzuspeichern.

10.4.17 Zugriffsrechte für Asset-Rubriken verwalten

Die Zugriffsrechte für Asset-Rubriken stellen Sie über die Rubriken-Verwaltung ein. Diese rufen Sie über die Funktion **Hauptmenü->Struktur->Rubriken** auf. Die Zuweisung der Berechtigungen können Sie wahlweise über den Dialog **Rubrik bearbeiten** oder das Rubriken-ACL-Werkzeug vornehmen. Die jeweilige Vorgehensweise ist eingehend in Abschnitt 6.1.4 **Rubriken-Zugriff** auf Seite 87 und Abschnitt 4.1.3 **Rubrikenrechte einstellen** auf Seite 58 beschrieben. Daher erfolgt hier nur eine Erklärung der einzelnen Berechtigungen.

Rollen, die diese Rubrik lesen dürfen, bzw. Lesen (r)

Diese Berechtigung benötigt eine Rolle, um Assets aus der betreffenden Rubrik sehen und in einem Dokument verwenden zu können. Soll auf eine Unterrubrik auch schreibender Zugriff möglich sein, müssen Sie auf die übergeordneten Rubriken ebenfalls das Leserecht erteilen.

Rollen, die Dokumente in dieser Rubrik anlegen dürfen, bzw. Dokumente erzeugen (c)

Rollen mit dieser Berechtigung können neue Assets hochladen. Dieses Recht beinhaltet auch das Anlegen von Asset-Varianten und die Änderung an den Metadaten eines Assets.

Rollen, die Dokumente in dieser Rubrik löschen dürfen, bzw. Löschen (d)

Nur Rollen mit dieser Berechtigung können Assets aus einer Rubrik entfernen oder verschieben.

Rollen, die diese Rubrik verändern dürfen, bzw. Schreiben (w)

Mit dieser Berechtigung können Benutzer die Rubriken-Einstellungen modifizieren sowie neue Unterrubriken anlegen. Diese Berechtigung bezieht sich nicht auf die in der Rubrik befindlichen Assets.

10.4.18 Listen-Ansicht konfigurieren

In der Listen-Ansicht des Media-Asset-Managements ist das Hinzufügen weiterer Spalten möglich. Zusätzlich hinzugefügte Spalten lassen sich auch wieder entfernen. Außerdem können Sie die Reihenfolge der angezeigten Spalten beeinflussen.

10.4.19 Neue Spalten hinzufügen und löschen

Um eine neue Spalte in der Listen-Ansicht hinzuzufügen, rufen Sie im MAM die Funktion **Ansicht->Spalten anzeigen->Spalte hinzufügen...** auf. Daraufhin öffnet sich folgender Dialog:



Abb. 10.3: Der Dialog Neue Spalte

Legen Sie die Einstellungen für die neue Spalte fest und speichern Sie diese durch einen Klick auf **Hinzufügen**.

ID

Tragen Sie hier den Namen des Asset-Metafelds ein, dessen Inhalt in der neuen Spalte angezeigt werden soll.

Name

Tragen Sie hier den Spaltentitel ein.

Sichtbar

Markieren Sie diese Checkbox, wenn die neue Spalte per Default sichtbar sein soll.

Sortierbar

Wenn Sie diese Option aktivieren, können die Benutzer die Listen-Ansicht nach dem Inhalt der neuen Spalte sortieren lassen.

Um eine zusätzlich hinzugefügte Spalte wieder zu entfernen, rufen Sie im MAM die Funktion **Ansicht->Spalten anzeigen->Anpassen...** auf. Sie gelangen zum Dialog "**Anpassen**":



Abb. 10.4: Der Dialog Anpassen

Klicken Sie auf den Listenpunkt vor der Spalte, die Sie entfernen möchten, ziehen Sie diesen mit gedrückter Maustaste über das Papierkorb-Icon im rechten Teil des Dialogfensters und lassen Sie dann die Maustaste los, um die Spalte zu löschen. Anschließend können Sie den Dialog schließen.



Hinweis:

Sie können nur nachträglich hinzugefügte Spalten entfernen.

10.4.8.2 Spaltenreihenfolge ändern

Um die Reihenfolge der angezeigten Spalten zu ändern, rufen Sie im MAM die Funktion **Ansicht->Spalten anzeigen->Anpassen...** auf. Sie gelangen zum Dialog "**Anpassen**" (Screenshot siehe oben). Die Spalten erscheinen in von oben nach unten gelesener Reihenfolge der Liste von links nach rechts eingeblendet in der Listen-Ansicht. Klicken Sie auf den Listenpunkt vor einem Spaltennamen, um die betreffende Spalte per Drag-and-Drop in der Liste nach oben oder unten zu verschieben.

10.5 Referenzen

Imperia behandelt Assets wie Dokumente. Sie lassen sich gesondert bearbeiten, publizieren und löschen. Daher ist es erforderlich, etwaige Verknüpfungen mit anderen Dokumenten zu verfolgen, um sicherzustellen, dass beispielsweise keine Grafiken gelöscht werden, auf die eines oder mehrere Dokumente referenzieren. Imperia hält bestehende Referenzen auf Assets nach und warnt Sie, wenn Sie versuchen sollten, ein referenziertes Asset zu löschen. Sie können sich die Referenzen auf ein Asset-Dokument im Media-Asset-Management anzeigen lassen. Je nach eingestellter MAM-Ansicht gehen Sie dazu wie folgt vor:

Iconbox-Ansicht

Klicken Sie auf das gewünschte Asset, um die zugehörigen Informationen anzuzeigen. Wählen Sie die Option **Referenzen zeigen**. Sie gelangen zum Dialog "**Asset-Referenzen**".

Listen-Ansicht

In der Listenansicht sind die Referenzen auf ein Asset über die Spalte **Referenzen** abrufbar. Eventuell müssen Sie diese Spalte erst anzeigen lassen (lesen Sie hierzu den Abschnitt "Spalten ein- und ausblenden (nur Listen-Ansicht)" im Benutzerhandbuch). Bei unreferenzierten Assets erscheint in dieser Spalte der Eintrag **Nicht referenziert**. Bestehen hingegen Referenzen auf ein Asset, erscheint dort der verlinkte Eintrag **Referenzen** mit einem Lupen-Icon. Klicken Sie auf den Link oder das Icon, um den Dialog "Asset-Referenzen" zu öffnen.



Abb. 10.5: Dialog Asset-Referenzen

Sie erhalten eine Übersicht aller Referenzen auf das Asset und seiner etwaigen Varianten. Beendete und in Bearbeitung befindliche Dokumente sind voneinander getrennt aufgelistet. Die Spalte **Dokument:** enthält die Namen der Dokumente, die auf das Asset zugreifen. Rechts daneben in der Spalte **Referenzierte Version:** ist die jeweils verwendete Variante des Assets angegeben.

Die Namen der Dokumente in der Liste sind mit dem entsprechenden Dokument verlinkt. Klicken Sie auf diesen Link, um eine Vorschau des Dokuments in einem neuen Fenster zu öffnen.

10.6 Einbau in ein Template

Damit ein Benutzer das Media-Asset-Management in einem Dokument verwenden kann, muss der Template-Code folgende Zeile enthalten:

```

<input name="IMPERIA:default1" type="text" />
```



Hinweis:

Die Angabe von width und height ist für den Aufruf des Media-Asset-Managements optional.

Weitere Einstellungsmöglichkeiten für die Integration des Media-Asset-Managements in Templates finden Sie im Abschnitt "Einbindung von Grafikdateien mit dem Media-Asset-Management" im Kapitel "Templates" im Programmierhandbuch.

10.7 Mehrfachupload mit Flash

Im Unterschied zum Standard-Mehrfachupload ermöglicht diese Variante dem Benutzer, mehrere Dateien innerhalb eines Verzeichnisses auf einmal für den Upload auszuwählen. Da die Implementierung auf Adobe Flash basiert, muss clientseitig eine möglichst aktuelle Version des Flash-Players installiert sein.

10.7.1 Kompatibilität

Der Imperia-Flashupload ist kompatibel zu folgenden Versionen des Flash-Players:

- Linux Flash v9.0.124.0 (Browser blockiert während des Uploads, siehe **Einschränkungen**)
- Mac Flash v9.0.115.0 und neuer (langsamer Upload bei Version 9.0.124.0)
- Windows Flash v9.0.115.0 und neuer

10.7.2 Einschränkungen

Durch einen Fehler im Flash-Player für Linux blockiert unter Linux der Browser beim Upload von Dateien und reagiert nicht mehr auf Benutzereingaben. Sobald der Upload-Vorgang beendet ist, ist diese Blockade beendet.

10.7.3 Rubrikeneinstellungen

Über die Rubriken-Metavariable `_imperia_swf_upload_template` stellen Sie das Template für den Flash-Multiupload ein. Geben Sie den Dateinamen des Templates ohne Endung oder etwaiges Präfix an. Die Template-Einstellung in den Rubrikeneinstellungen ist für den "normalen" Einzel- bzw. Mehrfachupload reserviert. Die relevanten Anweisungen für Templates für den Flash-Mehrfachupload sind im Abschnitt **Flash-Mehrfachupload (Media-Asset-Management)** unter **Syntax-Referenz** im Kapitel **Templates** im Programmierhandbuch eingehend beschrieben.

Der Workflow darf keine interaktiven Plug-Ins enthalten. Ausnahme ist hierbei das semi-interaktive Upload-Plug-In aus dem Standard-Workflow MAM. Sobald sich ein interaktives Plug-In im Workflow befindet, bricht der Flash-Upload an dieser Stelle ab.

Wenn Sie im MAM in eine so präparierte Rubrik wechseln, ist über den Menüpunkt **Datei** zusätzlich die Option **Multiupload (Flash)** verfügbar.

Kapitel 11. Der Daemon Hermes

Bei einem Daemon handelt es sich um einen Prozess, der ununterbrochen im Hintergrund ausgeführt wird und nach einem bestimmten Zeitplan oder aufgrund bestimmter System-Ereignisse definierte Funktionen ausführt.

Der Daemon von Imperia wird durch die Skripte `/site/bin/site_hermes_unix.pl` bzw. `/site/bin/site_hermes_win.pl` und `/site/bin/site_hermes.pl` repräsentiert und zeichnet sich durch einen modularen Aufbau aus. Alle Funktionen sind in externe Plug-Ins ausgelagert. Daraus ergibt sich ein Maximum an Flexibilität und Skalierbarkeit.

Er ist für alle automatischen Funktionen sowohl auf dem Produktionssystem als auch auf den Zielsystemen von Imperia verantwortlich. Dazu gehören die Überwachung von Verzeichnissen, die Ausführung von Systemdiensten und SiteActives.

Für Testzwecke ist es sinnvoll, den Daemon bei Bedarf zu starten. Er bietet zusätzlich einen Debug-Modus zur Fehlersuche, die Sie durch eine entsprechende Kommandozeilenoption aktivieren können.

Die Ausführungseigenschaften der einzelnen Funktionen der Plug-Ins werden in einer Tabelle gespeichert, die darüber Auskunft gibt, wann und unter welchen Bedingungen ein Plug-In ausgeführt wird.

11.1 Starten des Hintergrund-Daemons Hermes

Es empfiehlt sich, den Daemon im Hintergrund auszuführen. Unter Windows®-Betriebssystemen ist die zusätzliche Registrierung des Daemons als Systemdienst empfehlenswert. Der Hermes-Daemon muss auf dem Entwicklungssystem und auf den Zielsystemen laufen.

11.1.1 Starten des Daemons unter UNIX®

Unter UNIX® sollte Hermes im Hintergrund beim Starten des Systems ausgeführt werden. Um den Hintergrund-Daemon manuell unter UNIX® zu starten, gehen Sie wie folgt vor:

1. Öffnen Sie eine Konsole und wechseln Sie in das Verzeichnis `site/bin` Ihrer Imperia-Installation (Entwicklungs- bzw. Zielsystem).
2. Geben Sie dort folgenden Befehl ein:

```
nohup ./site_hermes_unix.pl &
```

11.1.2 Starten des Daemons unter Windows®

Um den Hintergrund-Daemon unter Windows® manuell zu starten, gehen Sie wie folgt vor:

1. Öffnen Sie eine DOS-Box und wechseln Sie in das Verzeichnis `site\bin` Ihrer Imperia-Installation (Entwicklungs- bzw. Zielsystem). Geben Sie folgenden Befehl ein:

```
perl site_hermes_win.pl
```

2. Lassen Sie die DOS-Box geöffnet.

11.1.3 Der Daemon als Systemdienst unter Windows®



Wichtig:

Alle benötigten Dateien finden Sie im Verzeichnis `site/util/XYNTService` der Imperia Installation.

11.1.3.1 Installation

Das Programm 'XYNTService' bietet die Möglichkeit, den Hermes als Systemdienst im Hintergrund unter Windows zu starten. Dafür ist die Anpassung der Datei 'XYNTService.ini' von Nöten, welche sich im Verzeichnis des Programms befinden muss. Anzupassen sind hier die Einträge 'CommandLine' und 'WorkingDir' um den Dienst an die Systemumgebung anzupassen.

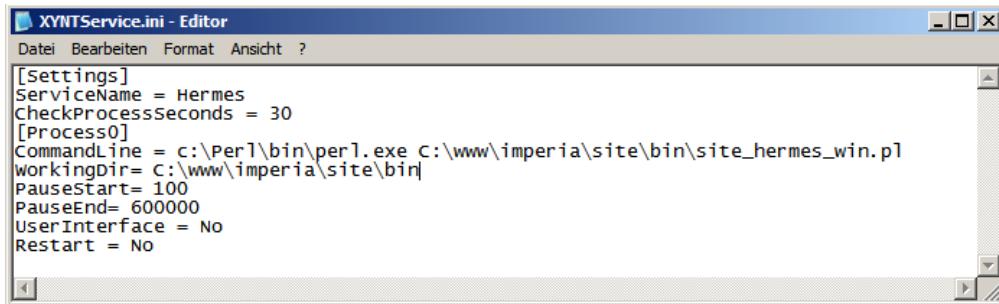


Abb. 11.1: Beispiel Konfiguration der XYNTService.ini

Weiterhin besteht die Möglichkeit, den Eintrag 'PauseEnd' nach Gebrauch zu verändern. 'PauseEnd' definiert die Zeit in Millisekunden, die beim Stop des Dienstes oder dem Herunterfahren des Systems dem Dienst (Hermes) verbleibt, um den Prozess ordentlich zu beenden. Im Beispiel werden 600000 Millisekunden gewählt, was 10 Minuten entspricht. Allerdings kann dieser Wert nicht ausreichend sein, falls das Hermes-Skript zu sehr ausgelastet ist. Verstreicht die Zeit, wird das Hermes-Skript vom Programm 'XYNTService.exe' umgehend beendet. Um den Systemdienst zu installieren, öffnen Sie eine DOS-Box, wechseln in das Verzeichnis, indem das Programm 'XYNTService.exe' liegt und geben folgenden Befehl ein:

```
XYNTService.exe -i
```

Wenn der Dienst erfolgreich installiert wurde, finden Sie im Windows Service Manager den Dienst 'Hermes' (der Name wird durch den Eintrag 'ServiceName' der 'XYNTService.ini' definiert) in der Liste der Dienste vor. Bei jedem Windows-Start wird der Dienst nun gestartet. Der Dienst kann auch manuell im Windows Service Manager oder durch folgenden Befehl einer DOS-Box gestartet werden:

```
net start Hermes
```

11.1.3.2 Deinstallation

Möchten Sie den Systemdienst entfernen, öffnen Sie eine DOS-Box, wechseln in das Verzeichnis, indem sich das Programm 'XYNTService.exe' befindet und geben folgenden Befehl ein:

```
XYNTService.exe -u
```

Wenn der Dienst erfolgreich destalliert wurde, ist der Eintrag 'Hermes' definiert durch den Eintrag 'ServiceName' der 'XYNTService.ini' in der Liste der Systemdienste im Windows Service Manager verschwunden.

11.2 Kommandozeilenoptionen

Über teilweise betriebssystemspezifische Kommandozeilenoptionen können Sie die Funktionsweise des Daemons beeinflussen.

11.2.1 Verwendung

```
site_hermes_unix.pl [OPTIONEN] | site_hermes_win.pl [OPTIONEN]
```

11.2.2 Optionen

Folgende Kommandozeilenoptionen sind, soweit nicht anders erwähnt, unabhängig vom Betriebssystem verfügbar:

Option *-b, --background*

Diese Option ist nur unter Unix® wirksam. Nach dem Start entkoppelt sich der Prozess vom Terminal und läuft im Hintergrund weiter.

Option *-d, --debug*

Diese Option aktiviert den Debug-Modus des Daemons. Es werden alle Aktivitäten des Daemons in der Log-Datei `hermes.log` im Verzeichnis <ServerRoot>/site/logs protokolliert.

Option *-e, --exit*

Verwenden Sie diese Option, damit der Daemon sich nach dem ersten Durchlauf automatisch beendet.

Option *-h, --help*

Diese Option zeigt einen kurzen Hilfetext in der Konsole/DOS-Box. Er enthält alle verfügbaren Kommandozeilenoptionen und eine kurze Beschreibung des Daemons. Dieser beendet sich sofort nach Anzeigen des Hilfetextes automatisch.

Option *-k restart*

Mit dieser nur unter Windows® verfügbaren Option veranlassen Sie einen Neustart des Hintergrund-Daemons. Das Skript entfernt vor dem Neustart automatisch die PID des laufenden Prozesses.

Option *-k stop / quit / exit*

Mit dieser nur unter Windows® verfügbaren Option beenden Sie den Hintergrund-Daemon. Das Skript entfernt dabei automatisch die PID des laufenden Prozesses.

Option *-p, --plug-in-debug*

Nutzen Sie diese Option, wenn Sie die Ausführung etwaiger Hermes-Plug-Ins debuggen möchten. Die Debug-Meldungen erscheinen in der Protokolldatei des Daemons `site/logs/hermes.log`.

Option *-s, -i, --seconds, --sleep=SECONDS*

Verwenden Sie diese Option, um eine Anzahl von Sekunden anzugeben, die der Daemon nach jedem Durchlauf pausieren soll.

Option *-t, --table*

Diese Option zeigt die Tabelle aller verfügbaren Plug-Ins. Der Daemon beendet sich sofort nach Anzeigen der Tabelle.

Option *-u USER, --user=USER*

Diese Option ist nur unter UNIX® wirksam. Wenn Sie den Daemon mit Root-Rechten starten, müssen Sie damit den User übergeben, unter dessen System-Konto der Daemon laufen soll. Aufgrund der UNIX®-Zugriffsrechte-Struktur muss der Daemon unter demselben User-Account laufen wie der Webserver. Dies ist häufig der User `nobody` oder auch `wwwrun`.

Option *-v, -z, --verbose*

Mit dieser Option aktivieren Sie die Ausgabe zusätzlicher Meldungen im Error-Log, die sonst nicht erscheinen.

11.2.3 Log-Datei

Alle Aktivitäten des Daemons werden in der Log-Datei `hermes.log` im Verzeichnis `site/logs` Ihrer Imperia-Installation protokolliert. Abhängig vom eingestellten Debug-Modus können Menge und Art der protokollierten Informationen stark variieren. Hier ein typischer Ausschnitt aus einer Log-Datei:

```
[Wed Jun 10 16:07:17 2007] HERMES IS RUNNING ...
[Wed Jun 10 16:07:18 2007] Hermes running with pid 3516
[Wed Jun 10 16:07:18 2007] FOUND 9 PLUGINS (SiteActive, ArchiveMinimizer,
    Import, ExpiryDate, MDBDel, CleanUp, MessageCheck, LogRotate, AutoPublish)
[Wed Jun 10 16:07:18 2007] SCHEDULING PLUGINS
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN SiteActive
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN ArchiveMinimizer
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN Import
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN ExpiryDate
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN MDBDel
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN CleanUp
[Wed Jun 10 16:07:18 2007] SUCCESSFULLY LOADED PLUGIN MessageCheck
[Wed Jun 10 16:07:19 2007] SUCCESSFULLY LOADED PLUGIN LogRotate
[Wed Jun 10 16:07:19 2007] SUCCESSFULLY LOADED PLUGIN AutoPublish
[Wed Jun 10 16:07:19 2007] STARTUP DONE

HERMES IS RUNNING ...
[Wed Jun 10 16:07:19 2007] [MessageCheck] Processing notification
    '469df2e40360bf2.free'
[Wed Jun 10 16:07:19 2007] [MessageCheck] Processing notification
    '469df2e403606e0.free'
[Wed Jun 10 16:07:19 2007] [MessageCheck] Processing notification
    '469df2e40360f41.free'
[Wed Jun 10 16:07:19 2007] [Import] looking for configuration file
    C:/imperia/dev/sanne/site/config/import.conf
[Wed Jun 10 16:07:19 2007] [Import] not found, falling back to hardcoded
    defaults
[Wed Jun 10 16:07:19 2007] [Import] successfully (re)initialized plugin
    Ascii
[Wed Jun 10 16:07:19 2007] [Import] successfully (re)initialized plugin
    ImperiaXML
[Wed Jun 10 16:07:19 2007] [Import] successfully (re)initialized plugin
    Meta
[Wed Jun 10 16:07:49 2007] SCHEDULING HERMES FOR QUIT ...
[Wed Jun 10 16:07:50 2007] EXIT SIGNAL RECEIVED.
[Wed Jun 10 16:07:50 2007] HERMES IS GOING AWAY...
[Wed Jun 10 16:07:50 2007] unlink
    C:/imperia/dev/sanne/site/logs/hermes.pid
[Wed Jun 10 16:07:50 2007] unlink
    C:/imperia/dev/sanne/site/logs/hermes.inst
[Wed Jun 10 16:07:50 2007] unlink
    C:/imperia/dev/sanne/site/logs/site_new_hermes_win.cmd
[Wed Jun 10 16:07:50 2007] TERMINATING PID: 2912
```



Hinweis:

Obenstehender Dateiauszug ist aus Darstellungsgründen umgebrochen.

Die Prozess-ID des Daemons wird zusätzlich in der Datei `hermes.pid` im Verzeichnis `site/logs` gespeichert. In der Regel existiert diese Datei nur, wenn der Daemon läuft. Sie wird gelöscht, sobald der Daemon normal beendet wird.

11.2.4 Plug-In-Tabelle

Beim ersten Start des Daemons werden alle gefundenen Plug-Ins in die Ausführungstabelle aufgenommen. Jede Funktion eines Plug-Ins erhält eine ID, anhand derer sie eindeutig identifiziert werden kann.

Die Tabelle enthält neben dem Namen (NAME), der Funktion (FUNCTION) und der ID (ID) alle weiteren Eigenschaften des Plug-Ins. Hierzu gehören:

die Ausführungszeit (EXEC_TIME),
die maximale Laufzeit (MAX_RUNTIME),
die Priorität (PRIORITY),
der Zeitpunkt der letzten Ausführung (LAST_RUN).

Die Einträge in der Plug-In-Tabelle werden nach Ausführungsriorität aufgelistet. Die Spalte der Ausführungszeit (EXEC_TIME) lässt mehrere Eintragsformate zu:

Zeitintervall in Sekunden (z. B.: 300 => 5 Minuten, o => immer),
fester Zeitpunkt in hh:mm[:ss] (z. B.: '0:00' => Mitternacht),
keine Ausführung aufgrund der Kennung no.

Die Tabelle bleibt nach der initialen Erstellung auch bei Beendigung des Daemons gültig.

Plug-Ins werden nur beim Start des Hintergrund-Daemons geladen. Um ein Plug-In zu entfernen, muss der Hintergrund-Daemon angehalten werden. Danach kann man das entsprechende Plug-In aus dem Verzeichnis site/modules/core/Dynamic/Hermes entfernen und den Hintergrund-Daemon wieder starten.

11.3 Hermes-Import

Für den Import von Daten mit Hilfe des Hintergrund-Daemons Hermes sind im Lieferumfang von Imperia mehrere Plug-Ins für verschiedene Formate enthalten. Der Hermes enthält eine fest programmierte Standard-Import-Konfiguration, mit der die Einstellungen für diese Standard-Formate festgelegt werden. Folgende Formate können importiert werden:

ASCII-Dateien mit der Endung .simple,
XML-Dateien, die nach dem Schema red*.xml benannt sind,
Imperia-Metadateien mit der Endung .meta.

Um andere Formate zu importieren, muss ein entsprechendes Import-Plug-In programmiert und die Standard-Import-Konfiguration erweitert werden. Die geänderte Konfiguration muss in der Datei /site/config/import.conf gespeichert werden.

Diese Datei existiert nach der Installation nicht, sondern muss angelegt werden, da die Standard-Import-Konfiguration fester Bestandteil des Moduls /site/modules/core/Dynamic/Hermes/Import.pm ist. Die in der Datei import.conf eingestellte Konfiguration überschreibt die Standard-Import-Konfiguration. Soll neben der neuen Konfiguration die Standard-Import-Konfiguration erhalten bleiben, muss diese aus dem Modul in die import.conf kopiert werden.

Alle importierten Dokumente enthalten nach dem Import eine zusätzliche Meta-Variable __imperia_imported, dessen Wert ein Zeitstempel ist, der die Anzahl der Sekunden seit einem bestimmten Zeitpunkt (Epoche) enthält. Dieser Zeitpunkt ist auf den meisten Systemen der 1. Januar 1970, 00:00 GMT.

11.3.1 Standard-Import-Konfiguration des Hermes-Daemons

Die Standard-Import-Konfiguration enthält Abschnitte für alle im Lieferumfang von Imperia enthaltenen Import-Plug-Ins. Jedes Format wird in der Konfigurationsdatei in einem filetype-Tag definiert. Dieses Tag muss folgende Tags enthalten:

- Ein mask-Tag, das einen Regulären Ausdruck enthält, der mit dem Namen der zu importierenden Dateien verglichen wird.
- Ein plugin-Tag, das den Namen des zu verwendenden Plug-Ins enthält.
- Ein optionales data-Tag, mit dem andere Daten an das Plug-In übergeben werden können.



Achtung:

Es ist nur ein data-Tag je filetype-Tag erlaubt. Weitere data-Tags werden ignoriert.

Die Standard-Import-Konfiguration ist wie folgt:

```
<?xml version="1.0" encoding="us-ascii"?>
<import_config>

<filetype>
```

```
<mask>^.*\simple$</mask>
<plugin>Ascii </plugin>
<data>zusätzliche Daten für das ASCII-Plug-In</data>
</filetype>

<filetype>
<mask>.*\red[^/]*\.xml$</mask>
<plugin>ImperiaXML </plugin>
<data>zusätzliche Daten für das ImperiaXML-Plug-In</data>
</filetype>
</filetype>
<filetype>
<mask>^.*\meta$</mask>
<plugin>Meta</plugin>
<data>zusätzliche Daten für das Meta-Plug-In</data>
</filetype>
</import_config>
```

Mit dieser Import-Konfiguration können folgende Formate importiert werden:

- ASCII-Dateien, die auf `.simple` enden und durch das ASCII-Plug-In verarbeitet werden.
- XML-Dateien, die auf den Regulären Ausdruck `.*\red[^/]*\.xml$` passen und durch das Imperia-XML-Plug-In verarbeitet werden.
- Imperia-Meta-Dateien, die auf `.meta` enden und durch das Meta-Plug-In verarbeitet werden.



Hinweis:

Das XML-Import-Format ist identisch mit dem von Imperia 7. Das Meta-Plug-In behandelt rohe Metainformationen, als wären sie mit Imperia 8 gespeichert worden.

Sollen andere Formate importiert werden, muss die Konfiguration angepasst werden. Für jedes weitere Format muss ein entsprechendes Plug-In vorhanden sein.

Zur Programmierung eigener Plug-Ins empfiehlt sich die Datei `/site/modules/core/Dynamic/Import/Ascii.pm` als Einstiegshilfe.

11.3.2 Zielrubrik bestimmen

Ein häufiger Grund für das Fehlschlagen des Imports ist das Fehlen der Rubrikinformationen, so dass Imperia nicht entscheiden kann, in welche Rubrik ein Dokument importiert werden soll.

Plug-Ins, die mit Imperia 8 ausgeliefert werden, entscheiden dies aufgrund des `data`-Tags in der Konfigurationsdatei, wobei versucht wird, die dort gefundenen Informationen zuerst der Meta-Variablen `__imperia_category` (Name der Rubrik) und, wenn dies fehlschlägt, der Meta-Variablen `__imperia_node_id` (Ziffernfolge) zuzuordnen.

Enthält die Konfigurationsdatei keine Informationen über die zu verwendende Rubrik, prüft Imperia, ob in den zu importierenden Dateien jeweils eine Rubrik angegeben ist. Ist dies nicht der Fall, schlägt der Import fehl und Hermes liefert eine Fehlermeldung.

Existieren im Rubrikenbaum mehrere Rubriken mit dem gleichen Namen, verwendet Hermes die erste passende Rubrik. Um eine bestimmte Rubrik zu verwenden, muss diese über die Node-ID eindeutig identifiziert werden.

Eine NodeID besteht aus zwei Teilen:

- der Rubrik-ID, zum Beispiel `/12/24/35/`
- der Dokument-ID, zum Beispiel `130336`.

Die vollständige NodeID aus diesen beiden Teilen stellt sich wie folgt dar: `/12/24/35/130336`. Sowohl die Dokument-ID als auch jeder einzelne Abschnitt der Rubrik-ID sind garantiert eindeutig, da Imperia sie aus einem automatisch inkrementierten Zähler generiert.

Enthält die Meta-Variable `__imperia_node_id` eine gültige NodeID, wird die Rubrik aus dieser Information gebildet, wobei die Dokument-ID nicht beachtet wird. Die Dateien werden dann in die gefundene Rubrik importiert.

11.3.3 Vorhandene Dokumente beim Import mergen

Neben der Möglichkeit, Dokumente anhand der NodeID in die richtige Zielrubrik zu importieren, gibt es zwei weitere Meta-Variablen, die dazu verwendet werden können, vorhandene und importierte Dokumente zu "mergen":

- `__imperia_clobber_by_name = 1 | 0`
- `__imperia_clobber_by_id = nodeID, z. B. /1/2/3`

Wenn Sie die Meta-Variable `__imperia_clobber_by_name` aktivieren, indem Sie sie auf `1` setzen, wird im Dokumenten-Baum nach einem Dokument gesucht, dass die gleiche URL wie das zu importierende Dokument hat. Wird ein solches Dokument gefunden, werden die beiden zusammengeführt, d. h. alle zusätzlichen Informationen aus der XML-Datei werden als Metafelder in Meta::Info des bereits im Archiv existierenden Dokumentes geschrieben.

Die Meta-Variable `__imperia_clobber_by_id` hat die gleiche Funktion, hierbei wird jedoch nicht nach einer URL, sondern nach einer NodeID gesucht. Diese Alternative ist schneller als die Suche nach der URL.

In beiden Fällen wird nur die ausgewählte Rubrik (siehe Kapitel 11.3.2) durchsucht.

11.3.4 Aufbau der Import-Daten

In diesem Abschnitt wird das jeweilige Format der Dateien beschrieben, die mit den Standard-Import-Plug-Ins importiert werden können.

11.3.4.1 ASCII-Import

Eine Datei, die durch das ASCII-Import-Plug-In importiert werden soll, muss aus Schlüssel-Wert-Paaren bestehen, wobei Schlüssel und Wert durch Gleichheitszeichen voneinander getrennt werden. Beispiel:

```
filename = index.html
directory = /import/ascii
title = Import-Test
metapage = movies
author = superuser
docid = 131875
__imperia_node_id = /4/10
expiry_date = 2002-03-11 10:06
```

11.3.4.2 XML-Import

Eine Datei, die mit Hilfe des Imperia-XML-Plug-Ins importiert werden soll, muss für jedes Metafeld ein entsprechendes Tag beinhalten, das wiederum den Wert des Metafeldes enthalten muss.

Beispiel:

```
<?xml version="1.0" encoding="ASCII" standalone="yes"?>
<IMPERIA_CONTENT>
  <filename>index.html</filename>
  <directory>/import/xml</directory>
  <title>Import-Test</title>
  <metapage>movies</metapage>
  <author>superuser</author>
  <docid>131876</docid>
  <__imperia_node_id>/4/11</__imperia_node_id>
  <expiry_date>2002-03-11 10:06</expiry_date>
</IMPERIA_CONTENT>
```

11.3.43 Meta-Import

Bei Imperia-Metadateien handelt es sich um Binärdateien, die nicht ohne Weiteres von Hand erzeugt werden können. Dieser Import wird dazu verwendet, bereits bestehende Imperia-Dokumente von einer Imperia-Installation in eine andere (auf Basis der gleichen Programm-Version) zu übertragen.

Kapitel 12. Die Imperia-Volltextsuche

12.1 Einführung

Die Suche ist integraler Bestandteil von Imperia und dient dem Auffinden beliebiger Wörter oder Wortkombinationen in Dokumenten eines Projekts und der strukturierten Ausgabe der Suchergebnisse. Sie dient dazu, den Benutzern eines Zielsystems eine Möglichkeit zum Durchsuchen der dort vorhandenen Dokumente zu bieten. Sie können damit auch die Dokumente auf dem Produktionssystem durchsuchen. Dies ist sinnvoll, wenn die Möglichkeiten der in die Imperia-Benutzeroberfläche integrierten Suche nicht ausreichen. Hierbei werden nur Dokumente indiziert, die den Workflow bereits verlassen haben.

Die Volltextsuche verarbeitet per Default ASCII-Dokumente (HTML, ASP ...). Mit zusätzlichen Hilfsprogrammen sind auch Microsoft-Office- Dokumente (*.doc, *.xls) und PDF-Dokumente durchsuchbar. Durch Gruppierungen kann die Suche auf bestimmte Bereiche eingeschränkt werden. Gruppierungen können auf Verzeichnis- oder auf Meta-Ebene erfolgen. Die Suche kann durch Verwendung unterschiedlicher Such- und Ergebnismasken für verschiedene Benutzergruppen angepasst werden. In den Ergebnislisten können konfigurationsabhängig beliebige Metainformationen angezeigt werden. Domains ermöglichen Ihnen die Verwaltung mehrerer Suchindizes. Auf diese Weise lassen sich beispielsweise die Datenbestände auf Systemen mit mehreren Mandanten sauber voneinander getrennt indizieren und durchsuchen.

Grundsätzlich indiziert die Volltextsuche den sichtbaren HTML-Text eines Dokuments. Es ist aber auch möglich, gezielt nicht sichtbare Metafeld-Inhalte zu indizieren und sichtbaren HTML-Code von der Indizierung auszuschließen.

12.2 Eingabe von Suchanfragen und Syntax für Suchbegriffe

Die Eingabe von Suchanfragen erfolgt über eigene Templates. Hinweise zur speziellen Syntax für die Erstellung von Suchtemplates finden Sie im Programmierhandbuch im Abschnitt "**Templates für die Imperia-Volltextsuche**". Auch für die Eingabe von Suchbegriffen gibt es einige Syntaxregeln, mit deren Hilfe sich Anfragen gezielter durchführen lassen. Ein Suchbegriff besteht aus einem oder mehreren Wörtern. Eine Verfeinerung der Suche und damit präzisere Suchergebnisse erzielen Sie durch die Verknüpfung mehrerer Suchwörter. Hierfür stehen Ihnen eine Gruppierungsfunktion, logische Operatoren und die Möglichkeit, gezielt nach Phrasen zu suchen, zur Verfügung. Außerdem können Sie bereits im Suchbegriff Ihre Anfrage nach dem Inhalt bestimmter Metafelder filtern. Die Syntax dieser speziellen Suchvarianten ist in den folgenden Abschnitten beschrieben. Der Abstand zwischen einzelnen Wörtern aus einem Suchbegriff im Text der gefundenen Dokumente spielt bei der Berechnung der Relevanz keine Rolle.

12.2.1 Groß- und Kleinschreibung

Die Imperia-Volltextsuche ignoriert die Groß- und Kleinschreibung von Suchbegriffen.

12.2.2 Sonderzeichen und Interpunktions

Die Volltextsuche ignoriert Sonderzeichen und Interpunktions generell, da diese generell nicht mit indiziert werden. Aus diesem Grund gibt es auch keine Ausnahmen für feststehende Begriffe mit Sonderzeichen wie z. B. 'C++'. Es gibt allerdings bei der Verwendung von Sonderzeichen in Suchbegriffen einige Ausnahmen. Diese sind "''", '(', ')', ':', '*' und '|'. Mit diesen Sonderzeichen führen Sie die oben angeführten speziellen Suchvarianten aus. Einzelheiten hierzu entnehmen Sie bitte den folgenden Abschnitten.

12.2.3 Wildcards

Verwenden Sie den Stern '*' in Suchbegriffen, um damit eine Wildcard für einen oder mehrere Buchstaben einzusetzen. Je mehr Wildcards eine Suchanfrage enthält, desto ressourcenintensiver ist deren Verarbeitung. Daher ist die Volltextsuche per Default so konfiguriert, dass pro Suchbegriff maximal eine Wildcard möglich ist. Bei dieser Einstellung dürfen Suchbegriffe mit Wildcard aus Performancegründen nur aus einem Wort bestehen. Beispielsweise liefert eine Suche nach 'a*' potenziell hunderte von Treffern. Diese mit weiteren per AND bzw. OR verknüpften Suchwörtern abzugleichen ist sehr ressourcenintensiv.

Verwenden Sie Wildcards bei der Default-Einstellung dennoch in Verbindung mit anderen Suchwörtern, erhalten Sie nicht die gewünschten Treffer. Die Volltextsuche ignoriert in diesem Fall die Wildcard. Eine Suche nach 'a*e AND Begriff' liefert statt Dokumenten mit Wörtern, die mit a beginnen und auf e enden und Begriff enthalten, Dokumente mit a oder (Default-Operator) e und Begriff. Wildcards, die nicht Bestandteil eines Suchworts sind (Leerzeichen davor und danach) ignorieren die Volltextsuche. Es ist also nicht möglich, eines oder mehrere Wörter in Suchphrasen (siehe Abschnitt 12.2.8 **Phrasensuche** auf Seite 158) durch eine Wildcard zu ersetzen.

Wenn Sie die Performance-Einbußen in Kauf nehmen und mehrere Wildcards pro Suchbegriff und einzelnem Suchwort sowie die Kombination von Suchwörtern mit und ohne Wildcards wünschen, können Sie dies in der Suchkonfiguration einstellen.

12.2.4 Alternative Schreibweisen / Worttrennungen

Eine Funktion zur Auffindung alternativer Schreibweisen für Komposita mit und ohne Bindestrich, getrennt und zusammengeschrieben ist derzeit nicht implementiert. Getrennt geschriebene Wörter werden auch getrennt indiziert. Es ist nicht möglich, gezielt nach Wörtern in Bindestrich-Schreibweise zu suchen.

12.2.5 Umlaute

Die Imperia-Volltextsuche unterscheidet bei Umlauten nicht zwischen den alternativen Schreibweisen. Die Suchbegriffe "Bürgermeister" und "Buergermeister" liefern also identische Ergebnisse.

12.2.6 Phonetisch ähnliche Alternativschreibweisen

Die Volltextsuche ignoriert phonetische Ähnlichkeit. Das gilt auch für Begriffe wie "Graphik" und "Grafik" etc.

12.2.7 Stopwörter

Die Imperia-Volltextsuche bietet die Möglichkeit, eine Liste von Stopwörtern zu definieren. Die Wörter aus dieser Liste erscheinen nicht im Index. Die Volltextsuche ignoriert Stopwörter außerdem in Suchanfragen. Eine Anfrage, die ein Wort aus der Stopwortliste sowie weitere Suchwörter enthält, liefert also dasselbe Ergebnis wie eine Anfrage ohne das Stopwort. Dabei ist unerheblich, ob die einzelnen Suchwörter mit **UND** oder **ODER** verknüpft sind oder ob es sich um eine Phrasensuche handelt. Besteht ein Suchbegriff ausschließlich aus Stopwörtern, liefert er gar keine Treffer.

Im Ergebnis-Template haben Sie die Möglichkeit, gesonderte Rückmeldung bei Verwendung etwaiger Stopwörter in Suchanfragen zu geben.

Stopwortlisten sind global oder domainspezifisch konfigurierbar. Kombinationen aus beiden Varianten sind ebenfalls möglich. Einrichtung und Verwaltung von Stopwortlisten sind in Abschnitt 12.5.3.7 **Stopwortlisten anlegen und verwalten** auf Seite 194 beschrieben.

12.2.8 Phrasensuche

Schließen Sie einen oder mehrere Begriffe in doppelte Anführungszeichen ein, um diese als Suchphrase zu kennzeichnen. Die Suche liefert daraufhin nur Treffer, die alle Wörter der Phrase in exakt der selben Reihenfolge enthalten. Die Interpunktions im Originaldokument ist dabei irrelevant.

12.2.9 Kategorisierte Suchanfragen

Die Einschränkung der Suchanfrage auf bestimmte Kategorien erreichen Sie durch die Definition von MetaGroups bei der Indexerstellung, die Sie dann im Suchformular zur Auswahl stellen. Einzelheiten hierzu finden Sie in Abschnitt 12.5.3.2 **Tags in Domain-Blöcken** auf Seite 168.

12.2.10 URL-Suche

Eine gezielte URL-Suche nach dem Muster der Google-Suchmaschine können Sie auch mit der Imperia-Volltextsuche umsetzen. Definieren Sie hierzu in den Dokumenten ein Metafeld namens site. Suchanfragen mit der Syntax 'site:www.myurl.de' liefern dann ein entsprechendes Ergebnis. Syntaktisch handelt es sich dabei um einen Filter nach Metafeldinhalten. Lesen Sie hierzu auch Abschnitt 12.2.13 **Nach Metafeldinhalten filtern** auf Seite 159.

12.2.11 Wortstammsuche

Momentan ist die Suche nach Alternativ-Wörtern mit dem gleichen Wortstamm nicht implementiert.

12.2.12 Gruppierung

Durch Einklammern "(")" können Sie eines oder mehrere Wörter in Suchbegriffen gruppieren. Achten Sie dabei auf korrekte Klammersetzung. Bei fehlenden öffnenden oder schließenden Klammern funktioniert die Gruppierung nicht. Stattdessen interpretiert die Volltextsuche die einzelnen Klammern als Bestandteile des Suchbegriffs.

12.2.13 Nach Metafeldinhalten filtern

Um Suchanfragen nach bestimmten Metafeldinhalten zu filtern, notieren Sie bei der Eingabe des Suchbegriffs Metafeldname:Begriff bzw. Metafeldname:[Begriff Begriff2]. Das zweite Beispiel mit den Pipe-Zeichen beinhaltet eine implizite UND-Verknüpfung der beiden Suchbegriffe. Es erscheinen nur Suchergebnisse, auf die der definierte Filter zutrifft, in denen es also das betreffende Metafeld mit dem passenden Inhalt gibt.

12.2.14 Suchwörter verknüpfen und ausschließen, logische Operatoren

Mit den Operatoren AND und OR verknüpfen Sie Teile des Suchbegriffs miteinander. Die Verknüpfung 'Wort1 AND Wort2' liefert Ihnen nur Treffer, in denen beide Suchwörter enthalten sind. Bei 'wort1 OR Wort2' erhalten Sie zudem Treffer, in denen eines der beiden Suchwörter enthalten ist. Wollen Sie bestimmte Wörter aus der Suche ausschließen, nehmen Sie diese mit einem vorangestellten NOT in den Suchbegriff auf. Die Groß- und Kleinschreibung ist bei den drei Operatoren AND, OR und NOT irrelevant. In der Suchkonfiguration (Datei site/config/fts.conf) können Sie sprachspezifische Synonyme für die logischen Operatoren angeben. Außerdem funktionieren folgende Sonderzeichen als "Shortcuts" für logische Operatoren:

Operator	Shortcut
AND	+
OR	?
NOT	-, !

Tabelle 12.1. Shortcuts für logische Operatoren

Logische Operatoren lassen sich beliebig mit Gruppierungen kombinieren. Mehrere Suchwörter ohne Operator im Suchbegriff verknüpft die Volltextsuche per Default mit OR. Diesen Default können Sie in der Suchkonfiguration ändern (siehe Abschnitt 12.5.3.4.3 **Direktiven in Option-Blöcken** auf Seite 185, Direktive defaultOperator).

Eine Besonderheit stellen zusammengesetzte Suchworte mit Bindestrich dar. So wird bei der Suche nach "Handball-WM" intern folgendes gesucht: (handball OR wm) OR "handball-wm" OR "handballwm". Ist als Standardoperator "AND" gesetzt, so wird folgendes gesucht: (handball AND wm) OR "handball-wm" OR "handballwm".

12.3 Anzeige von Suchergebnissen

Auch die Darstellung von Suchergebnissen erfolgt über eigene Templates. Die spezielle Syntax für die Erstellung von Ergebnistemplates ist im Programmierhandbuch im Abschnitt "**Templates für die Imperia-Volltextsuche**" erklärt. Die Sortierung der Trefferliste erfolgt per Default absteigend nach Relevanz. Weitere Sortieroptionen lassen sich durch die Konfiguration von Ergebnisfiltern aktivieren. Lesen Sie hierzu Abschnitt 12.3.3 **Ergebnisfilter** auf Seite 160, bzw. Abschnitt 12.5.3.4.6 **UserSort: Sortierung der Treffer bestimmen** auf Seite 191 und die darin enthaltenen Erläuterungen zur Direktive sortBy. Bei der Anzeige der einzelnen Treffer haben Sie die Möglichkeit, eine frei konfigurierbare Menge Kontext anzuzeigen, in dem der gefundene Suchbegriff hervorgehoben erscheint.

12.3.1 Kontextanzeige

Bei der Anzeige von Treffern gibt die Volltextsuche für jeden Treffer einen Textausschnitt mit dem Suchbegriff aus. Die Wortanzahl für diesen Textausschnitt können Sie bestimmen, nicht jedoch die Position des Suchbegriffs in dem Ausschnitt. Bei aus mehreren Wörtern zusammengesetzten Suchbegriffen erscheint der erste Textabschnitt aus einem gefundenen Dokument, in dem die meisten der gesuchten Wörter vorkommen und der die vorgegebene Wortanzahl nicht überschreitet. Selbst wenn das Dokument alle gesuchten Wörter enthält, müssen diese nicht unbedingt alle in dem dargestellten Kontext erscheinen, wenn die eingestellte Wortanzahl dafür nicht ausreicht. Auch muss bei einer Phrasensuche nicht zwangsläufig der erste Abschnitt im Dokument der sein, der die gesuchte Phrase enthält. Sollten im selben Dokument vorher alle gesuchten Wörter in einem darstellbaren Textabschnitt in anderer Reihenfolge vorkommen, zeigt die Volltextsuche diesen an.

12.3.2 Highlighting

Bei der Ausgabe der Treffer lassen sich Suchbegriffe hervorgehoben darstellen. Die hierfür notwendigen Anweisungen tragen Sie im Ergebnistemplate ein. Für die Hervorhebung in Suchbegriffen mit Wildcards stehen zudem spezielle Hervorhebungsmodi und CSS-Klassen zur Verfügung. Lesen Sie hierzu den Abschnitt "**Templates für die Imperia-Volltextsuche**" im Programmierhandbuch. Imperia enthält eine Standard-CSS-Datei für die Darstellung von Hervorhebungen. Durch die Definition einer lokalen CSS-Klasse im Ergebnistemplate können Sie dieses Standard-Design der Hervorhebung durch eigene Formatierungen ersetzen.

12.3.3 Ergebnisfilter

Normalerweise schickt Imperia direkt nach der Verarbeitung der Suchanfrage die Ergebnisliste an den anfragenden Client zurück. Eine offene Plug-In-Schnittstelle ermöglicht es, die Trefferliste vor der Darstellung aufzubereiten. So lassen sich beispielsweise spezielle Sortiermethoden oder auch Manipulationen an der Darstellung der Metainformationen der gefundenen Dokumente implementieren.

Ergebnisfilter-Plug-Ins legen Sie im Verzeichnis `site/modules/core/Dynamic/FTSResultFilter` Ihres Systems ab. Dort gibt es auch schon einige Beispiele. Die verfügbaren Methoden für die Aufbereitung der Trefferliste sind im POD des Moduls `site/modules/core/Imperia/FTS/ResultFilterPlugin.pm` beschrieben.

Für Hinweise zur Aktivierung von Ergebnisfiltern in der Volltextsuche lesen Sie bitte Abschnitt 12.5.3.4.3, „`resultFilterPlugin`“.

12.4 Berechnung der Treffer-Relevanz und Boni

Die Berechnung der Relevanz eines Dokuments in einer Trefferliste erfolgt in mehreren Teilschritten. Bei der Indizierung ermittelt die Volltextsuche für jedes im Text enthaltene Wort einen Relevanzwert. Dieser ergibt sich daraus, in wie vielen Dokumenten das betreffende Wort jeweils wie häufig vorkommt. Die endgültige Relevanz der einzelnen Wörter im Index ergibt sich dann aus der Relation zum Durchschnittswert. Wenig aussagekräftige Wörter wie "und", "ist" etc. erhalten so eine vergleichsweise geringe Relevanz, da sie in sehr vielen Dokumenten sehr häufig vorkommen.

Bei einer Suchanfrage ermittelt die Volltextsuche die Relevanz der gefundenen Dokumente aus dem Produkt der Relevanz des Suchbegriffs und seiner Häufigkeit im jeweiligen Dokument. Besteht der Suchbegriff aus mehreren Wörtern, errechnet sich die Relevanz der Dokumente aus der Summe der Werte der einzelnen Wörter. Die Trefferrelevanz der Dokumente können Sie durch die Vergabe von Boni in zweierlei Hinsicht beeinflussen: Boni lassen sich anhand der Position eines Wortes im Dokument und datumsbezogen definieren. Lesen Sie hierzu Abschnitt 12.5.3.3.1, „`<bonus>`“ und Abschnitt 12.5.3.4.3 **Direktiven in Option-Blöcken** auf Seite 185, Direktiven zu `bonus_date`.

Auch wenn der letztendlich ermittelte Relevanzwert eine absolute Zahl ist, erfolgt die Ausgabe als Prozentangabe. Dabei dient das Dokument mit dem höchsten Wert als Ausgangspunkt für die Umrechnung.

12.5 Einrichten der Imperia-Volltextsuche

Die Volltextsuche verwendet zwei Skripte: `site/bin/fts_index.pl` für die Indexerstellung und `cgi-bin/fts_search.pl` für die Verarbeitung der Suchanfragen.

Außerdem gibt es das Hilfsskript `fts_conf_convert.pl` zum Konvertieren einer bestehenden Konfiguration älterer Versionen der Imperia-Volltextsuche in das neue Format. Die gesamte Konfiguration wird über die Konfigurationsdateien `/site/config/index.conf` und `/site/config/fts.conf` gesteuert. Mit `/site/config/index.conf.sample` existiert eine Beispielkonfiguration, in welcher die Funktion jedes Parameters kurz erläutert wird. Auch solche, auf welche in dieser Dokumentation nicht explizit eingegangen wird.

Eine Anleitung zum schnellen Start finden Sie im Abschnitt 12.5.2 **Schnellstartanleitung** auf Seite 161. Auf die einzelnen Skripte wird in späteren Abschnitten detailliert eingegangen.

12.5.1 Voraussetzungen

Die Imperia-Volltextsuche benötigt folgende Perlmodule:

- BerkeleyDB >= 4.0. (empfohlene Version, mindestens jedoch Version 3.0)
- BerkeleyDB::Btree
- Config::General (*)
- File::Spec (*)
- File::Copy (*)
- File::Path (ab Perl 5.8.X Bestandteil des Interpreters)
- Tree::RedBlack (*)

Die mit einem Stern gekennzeichneten Module sind in der Imperia-Distribution enthalten, die übrigen müssen Sie vor dem Einsatz der Volltextsuche selbst installieren. Diese Perl-Module können Sie kostenlos von www.cpan.org beziehen. Sollte eines der benötigten Module fehlen, bekommen Sie dies durch eine entsprechende Fehlermeldung angezeigt.

Ein optionales Modul ist folgendes:

- Compress::Zlib

Dieses Modul brauchen Sie nur, wenn Sie den Cache des Index aus Platzersparnisgründen komprimieren möchten.

Wenn Sie neben HTML- und einfachen Textdateien andere Dateitypen indizieren und durchsuchen wollen, z. B. PDF-Dokumente, benötigen Sie zusätzliche Parser-Programme, die Sie nachträglich installieren müssen.

Gegenwärtig gibt es folgende Parser-Plug-Ins :

Für Microsoft Office Dokumente das Programm Catdoc, zu beziehen über <http://www.45.free.net/~vitus/software/catdoc/> (Stand der Quellenangabe: April 2007). Benötigt wird die Version 0.94x, die unter der GPL 2 Lizenz steht. Es handelt sich dabei um eine Suite von Parsern, zu der folgende Programme gehören:

- DOC -Parser für Word-Dokumente
- XLS -Parser für Excel-Dokumente
- PPT -Parser für Powerpoint-Dokumente

Für PDF-Dateien brauchen Sie XPDF, einen Parser, der auf einer Programmsuite namens XPDF basiert. Der Parser ist unter den folgenden URLs erhältlich: <http://www.xpdf.com/> (Stand der Quellenangabe: Juni 2009) und <http://www.foolabs.com/xpdf/>. Dazu gehören zwei Programme:

- pdftotext
- pdfinfo

12.5.2 Schnellstartanleitung

Das Einrichten der Imperia-Suche setzt sich aus den drei Schritten Konfiguration, Indizierung und Template-Erstellung zusammen. Durch die Konfiguration wird unter anderem festgelegt, welche Dokumente über die Suche auffindbar sein sollen. Templates dienen zur Eingabe von Suchbegriffen und Suchparametern sowie zur Darstellung der Suchergebnisse. Bei der Indizierung wird ein Suchindex erstellt. Die eigentliche Suche

beschreibt, in welcher Weise gesuchte Worte miteinander verknüpft werden können. Für alle konfigurierbaren Parameter gibt es Default-Werte. Diese kommen auch in den Konfigurationsbeispielen zum Einsatz. Im Verzeichnis `site/fts/templates` Ihrer Imperia-Installation liegen Beispiel-Templates für Suchanfragen und die Anzeige der Suchergebnisse. Wenn Sie also auf eine besondere Konfiguration verzichten können, sind nur wenige Schritte notwendig, damit Sie die Volltextsuche einsetzen können:

1. Konfiguration: Im Verzeichnis `site/config` finden Sie Beispiel-Konfigurationsdateien für die Indexerstellung (`index.conf.sample`) und die Ausgabe der Suche (`fts.conf.sample`). Kopieren Sie diese Dateien nach `index.conf` und `fts.conf`.
2. Indexerstellung: Führen Sie das Script `site/bin/fts_index.pl` mit dem Parametern `-b` und `-i` aus, um den Index zu erstellen.



Hinweis:

Bei umfangreichen Projekten kann dieser Vorgang einige Zeit in Anspruch nehmen.

3. Jetzt können Sie Suchanfragen in die vorbereiteten Templates eingeben, indem Sie das Skript `fts_search.pl` aus dem Verzeichnis `cgi-bin` im Browser aufrufen: http://ihr_server.de/cgi-bin/fts_search.pl bzw. http://ihr_server.de/cgi-bin/fts_search.pl?ADV=1 für ein verfeinertes Suchformular mit zusätzlichen Optionen für die Suche.

12.5.3 Konfiguration der Volltextsuche

Die Imperia-Volltextsuche gliedert sich in zwei Bereiche - die Erstellung eines Suchindex und die Suche nach Begriffen im Suchindex inklusive der Anzeige der gefundenen Treffer.

Diese funktionale Trennung spiegelt sich auch in der Konfiguration wider, die in zwei getrennten Dateien vorgenommen wird. Bereits bestehende Suchkonfigurationen aus älteren Imperia-Versionen können mit einem Skript migriert werden. Lesen Sie hierzu Abschnitt 12.5.3.3 **Die Konfiguration der Indexerstellung (index.conf)** auf Seite 163 . Wenn Sie mehrere Indizes verwalten wollen, können Sie dies über die Definition mehrerer Domänen erreichen. Das ermöglicht beispielsweise die Nutzung einer Suchkonfiguration für mehrere Zielsysteme.

Domänen können Sie auch nutzen, um unterschiedliche Ausschnitte eines Datenbestandes getrennt von einander durchsuchbar zu machen. Wenn Sie beispielsweise einen Teil der auf dem Zielsystem verfügbaren Daten nur für Ihre Mitarbeiter durchsuchbar machen wollen, müssen Sie entsprechende Domänen konfigurieren. In den meisten Fällen reicht es jedoch aus, die Grundeinstellung mit einer Domain ("default") zu verwenden.

12.5.3.1 Allgemeine Syntaxkonventionen

Die Syntax der Anweisungen zur Konfiguration der Volltextsuche ist an die Syntax der Konfiguration des Apache Webservers angelehnt, um Ihnen einen schnelleren Einstieg zu ermöglichen. Einige Anweisungen werden in spitzen Klammern notiert. Diese "Tags" bilden Blöcke, in denen Sie weitere Tags oder Direktiven notieren können, um beispielsweise eine Reihe von Parametern für Verzeichnisse festzulegen. Ähnlich wie bei Markup-Sprachen oder auch bei der Apache-Konfiguration gibt es dabei ein einleitendes und ein abschließendes Tag.

Beispiel:

```
<option>
  Direktiven
</option>
```

Bei manchen Tags wird im einleitenden Tag bereits ein Parameter festgelegt. Dieser kann durch ein optionales Gleichheitszeichen und durch Anführungsstriche, deren Verwendung ebenfalls optional ist, besser kenntlich gemacht werden. Beispiel:

```
<directory = "pfad/zum/verzeichnis">
  Direktiven
</directory>
```

bzw.

```
<directory pfad/zum/verzeichnis>
  Direktiven
</directory>
```

Außerdem gibt es noch Direktiven, die in der Form

```
Direktive="Wert"
```

notiert werden und einen Wert für einen bestimmten Parameter bestimmen.

Beispiel:

```
chmap = "UTF-8"
```

Auch hier ist Ihnen die Verwendung von Gleichheitszeichen und Anführungszeichen freigestellt. Eine Ausnahme bilden sowohl für Tags als auch für einfache Direktiven die Fälle, in denen der beschriebene Parameter bzw. der gesetzte Wert Leerzeichen enthält. Dort ist die Verwendung von Anführungszeichen bindend, um sicherzustellen, dass der komplette Ausdruck gewertet wird.

Allgemein ist der Einsatz von Anführungszeichen jedoch empfehlenswert, um Fehlinterpretationen zu vermeiden. Das oben stehende Beispiel würde nämlich mit einer winzigen Änderung nicht mehr funktionieren.

Falsch ist:

```
<directory pfad/zum/verzeichnis >
  Direktiven
</directory>
```

Das Leerzeichen vor der schließenden spitzen Klammer interpretiert der Parser als Bestandteil der Pfadangabe, also "pfad/zum/verzeichnis " statt "pfad/zum/verzeichnis". Im Gegensatz dazu funktioniert die Variante mit Anführungsstrichen problemlos.

Richtig:

```
<directory "pfad/zum/verzeichnis" >
  Direktiven
</directory>
```

In diesem Fall ignoriert der Parser das überflüssige Leerzeichen einfach. Bei den Tags und Direktiven der Konfigurationsdateien wird Groß- und Kleinschreibung grundsätzlich nicht berücksichtigt. Die Codebeispiele in den folgenden Abschnitten verwenden allerdings durchgängig die Kleinschreibweise bzw. die "CamelCase"-Schreibweise.

Kommentarzeilen leiten Sie durch ein Raute-Zeichen (#) am Zeilenanfang ein. Verwenden Sie für Wertzuweisungen keine Umlaute oder Sonderzeichen außer Bindestrich (-) und Unterstrich (_). Ausgenommen hiervon sind reguläre Ausdrücke.

12.5.2 Reguläre Ausdrücke in der Konfiguration

Reguläre Ausdrücke in den Konfigurationsdateien sind in ihrer Funktionalität stark an Perl angelehnt. Ihnen stehen sämtliche syntaktischen Möglichkeiten, die Sie in Perl im "Rumpf" eines Suchmusters (zwischen den Schrägstrichen) notieren können, zur Verfügung. Modifier (single line, multi line, ignore case etc.) sind nicht verwendbar. Generell werden die von Ihnen definierten Muster mit der Option "ignore case" ausgewertet. Einen Sonderfall stellen Datei- bzw. Verzeichnisnamen dar. Hier ist je nach Betriebssystem die Groß- und Kleinschreibung relevant.

12.5.3 Die Konfiguration der Indexerstellung (index.conf)

Über die Datei site/config/index.conf konfigurieren Sie die Indexerstellung. Die Konfiguration wird vom Skript fts_index.pl eingelesen und verarbeitet.



Wichtig:

Wichtig: Änderungen an der Konfiguration der Indexerstellung werden erst aktiv, wenn Sie einen neuen Index erstellen.

Im Folgenden finden Sie eine Auflistung der verwendeten Tags und Direktiven. Die einzelnen Anweisungen sind dabei jeweils in der höchstmöglichen Strukturebene aufgeführt, in der sie verwendbar sind. Einige können Sie auch noch an anderen Stellen einsetzen, was dann entsprechend vermerkt ist.

12.5.3.1 Basis-Tags und -Direktiven

<domain>

Das Tag „domain“ beschreibt verschiedene unabhängige Konfigurationen für verschiedene Indizes innerhalb einer Konfigurationsdatei. Bis auf wenige Ausnahmen, die allgemeingültige Einstellungen festlegen und deswegen auch auf oberster Ebene definiert werden können, notieren Sie alle weiteren Tags und Direktiven innerhalb eines Domain-Abschnittes.

Die Verwendung von Domains ist flexibel. Sie können entweder verschiedene Datenbestände oder auch ein- und denselben Datenbestand mit unterschiedlichen Einstellungen indizieren. Beispielsweise lassen sich unterschiedliche Domains für Mitarbeiter und Kunden einrichten. Bei der Indexerstellung für die Suche, die von Ihren Kunden genutzt werden soll, würden Sie dann die Bereiche mit Informationen ausklammern, die nur zum firmeninternen Gebrauch bestimmt sind. Der Index für die von den Mitarbeitern genutzte Suche würde jedoch auch die nicht-öffentlichen Bereiche einschließen.

Gegenüber einer Filterung, die Sie zum Beispiel durch Gruppierung der Information erreichen, hat dies den Vorteil, dass die Trennung der verfügbaren Daten sicherer ist. In Informationsgruppen „versteckte“ Inhalte sind demgegenüber durch geschickte bzw. glückliche Manipulation der Aufrufparameter des Suchskripts unter Umständen auch unberechtigten Besuchern zugänglich.



Hinweis:

Die in der Datei index.conf.sample genannte domain „default“ ist eine eigenständige Domain und dient nicht etwa der Erfassung von Default-Werten für andere Domains.

Syntax:

```
<domain= "domain_name">
  [Tags und Direktiven zur Konfiguration des Index dieser Domain]
</domain>
```

Beispiel:

```
<domain = "secure">
  #limited index only intranet
</domain>
<domain = "web">
  #general index public web content
</domain>
```

<settype>

Mit dem Tag „settype“ definieren Sie, welche Dateitypen indiziert werden sollen. Der Aufbau entspricht einer mehrwertigen Liste. Folgende Schlüssel stehen zur Verfügung:

- HTML

Mit diesem Plug-In können Sie HTML-Dateien indizieren.

- TXT

Dieses Plug-In ermöglicht die Indizierung von TXT-Dateien.

- XPDF

Mit Hilfe des XPDF-Plug-Ins ist es möglich, PDF-Dateien zu indizieren.

- DOC

Um DOC-Dateien zu indizieren, verwenden Sie bitte dieses Plug-In.

- PPT

Das PPT-Plug-In erlaubt die Indizierung von PPT-Dateien.

- XLS

Unter Verwendung des XLS-Plug-Ins lassen sich XLS-Dateien indizieren.

- NULL

Mit Hilfe des NULL-Plug-Ins können Sie einen reinen Meta-Index erstellen.

Syntax:

```
<settype>
  Typ = [regulärer Ausdruck]
</settype>
```

Der reguläre Ausdruck beschreibt Dateiendungen.

Beispiel:

```
<setType>
  HTML = "\..?html?\$"
  TXT = "\.txt\$"
</setType>
```



Hinweis:

Die Dokumenttypen PDF, DOC, PPT und XLS werden über externe Parser indizierbar gemacht. Dieses Tag kann außerdem innerhalb eines Domain-Abschnittes eingesetzt werden.

Dieses Tag ist auch in Domain-Blöcken anwendbar.

<bonus>

Mit dem Bonus-Tag können Sie die Gewichtung der Volltextsuche beeinflussen, indem Sie einem Begriff abhängig von seiner Position in der Struktur eines Dokuments zusätzliches Gewicht verleihen. Dabei vergeben Sie Boni für bestimmte Strukturelemente und Metafelder eines Dokuments. Diese werden dann bei der Ermittlung der Häufigkeit eines Begriffes in einem Dokument hinzugezählt, wenn ein Begriff innerhalb des entsprechenden Strukturelements vorkommt.

Wenn Sie beispielsweise für Überschriften ersten Grades einen Bonus von zwei eingestellt haben, erhöht sich der Zähler für einen Begriff also jedes Mal um drei, wenn dieser innerhalb einer Überschrift ersten Grades erscheint (+1 für den Begriff, +2 durch den Bonus).

Sie können für folgende Elemente Boni vergeben:

- title (<title></title>)
- bold ()
- italics (<i></i>)
- header1 (<h1></h1>)
- header2 (<h2></h2>)
- header3 (<h3></h3>)

- header4 (<h4></h4>)
- headers5 (<h5></h5>)
- header6 (<h6></h6>)
- keywords (Imperia-Metafeld oder pageMetaComment keywords)
- description (Imperia-Metafeld pageMetaComment description)



Wichtig:

Die obige Auswahl von Strukturelementen steht Ihnen nur bei HTML-Dokumenten in vollem Umfang zur Verfügung. Bei PDF-Dateien kann lediglich für das Title-Element ein Bonus vergeben werden. Für Microsoft Office®-Dokumente lassen sich keine Boni festlegen.

Als Werte kommen Ganzzahlen und Dezimalzahlen in Frage. Notieren Sie Dezimalzahlen in der englischen Schreibweise mit einem Punkt als Dezimalzeichen.

Syntax:

```
<bonus>
  Strukturelement = Ziffer
</bonus>
```

Beispiel:

```
<bonus>
  italics = 1.2
  title = 2
  bold = 1.3
  header1 = 1.2
  header2 = 1.2
  header3 = 1.2
</bonus>
```

Sie haben außerdem die Möglichkeit, Boni domainspezifisch zu definieren.

<exclude>

Sie können dieses Tag nutzen, um den Inhalt einzelner Metafelder oder Dateien, die ein bestimmtes Metafeld enthalten, von der Indizierung auszuschließen.

Syntax:

```
<exclude>
  [Direktiven]
</exclude>
```

Eine Beschreibung der in Exclude-Blöcken verfügbaren Direktiven und Beispiele finden Sie in Abschnitt 12.5.3.3.3 **Direktiven in Exclude-Blöcken** auf Seite 171. Dieses Tag lässt sich auch innerhalb von Domain-, Directory-, DirectoryMatch- und Files-Blöcken einsetzen.

<include>

Sie können dieses Tag nutzen, um Dateien, die ein bestimmtes Metafeld enthalten, indizieren zu lassen.

Syntax:

```
<include>
  [Direktiven]
</include>
```

Eine Beschreibung der in Include-Blöcken verfügbaren Direktiven und Beispiele finden Sie in Abschnitt 12.5.3.3.4 **Direktiven in Include-Blöcken** auf Seite 171.

Dieses Tag lässt sich auch innerhalb von Domain-, Directory-, DirectoryMatch- und Files-Blöcken einsetzen.

<pageMetaComment>

Mit diesem Tag können Sie ein Metafeld definieren, das durchsucht werden und dessen Inhalt in Templates ausgegeben werden kann. Innerhalb des Tags legen Sie eine Start- und eine Endmarkierung fest, mit denen Sie den Metafeldinhalt vom übrigen Text des Dokuments abgrenzen.

Syntax:

```
<pageMetaComment = "meta_name">
  start = "<!--startcomment-->"
  stop = "<!--stopcomment-->"
</pageMetaComment>
```

Beispiel:

```
<pageMetaComment = "FSK 0">
  start = "<!--start_FSK 0-->"
  stop = "<!--stop_FSK 0-->"
</pageMetaComment>
```

Dieses Tag ist auch in Directory-, DirectoryMatch- und Files-Blöcken verwendbar.

blacklist

Mit dieser Direktive geben Sie eine oder mehrere Stopwortlisten an, die für alle konfigurierten Domains der Volltextsuche gültig sein sollen. In einer Stopwortliste aufgeführte Wörter indiziert die Volltextsuche nicht. Damit eine Suchanfrage die Stopwörter ebenfalls ignoriert, müssen Sie die Direktive auch in der Suchkonfiguration `site/config/fts.conf` setzen.

Syntax:

```
blacklist = BEZEICHNER
```

Mit `BEZEICHNER` geben Sie eine Stopwortliste an, die Sie mit dem Skript `site/bin/fts_blacklist.pl` in das System eingetragen haben. Lesen Sie hierzu auch Abschnitt 12.5.3.7 **Stopwortlisten anlegen und verwalten** auf Seite 194.

Wenn Sie mehrere Stopwortlisten einsetzen möchten, verwenden Sie die Direktive mehrfach, für jede Stopwortliste einmal.

Diese Direktive können Sie auch innerhalb von Domain-Blöcken verwenden, dann gilt die Stopwortliste nur für die betreffende Domain.

PDFINFO

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

PDFTOTEXT

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

XLS2CSV

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

CATDOC

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

CATPPT

Sollte sich das Parserprogramm nicht im Umgebungspfad befinden, notieren Sie eine absolute Pfadangabe zum Programm. Dieses Tag kann auch innerhalb von Domain-Blöcken verwendet werden.

125332 Tags in Domain-Blöcken

<metaGroup>

Dieses Tag erlaubt Ihnen, Dateien anhand von Metafeldern und deren Inhalt zu Gruppen zusammenzufassen. Diese Gruppen können Sie bei Suchanfragen als Filter an das Suchskript übergeben (lesen Sie hierzu den Abschnitt „Übergabeparameter und Variablen in Suchtemplates“ im Kapitel „Templates für die Volltextsuche“ im „Programmierhandbuch“). In einem MetaGroup-Block definieren Sie eines oder mehrere Metafelder mit zugehörigen Inhalten. Dokumente, die die entsprechenden Metafelder und Metafeldinhalte aufweisen, gehören dann zu dieser Gruppe.

Die eigentliche Definition der einzelnen Metafelder und jeweiligen Gruppen nehmen Sie im Tag `<meta>` vor, das in Abschnitt 12.5.3.3.7 **Tags und Direktiven in MetaGroup-Blöcken** auf Seite 179 beschrieben wird. Dort finden Sie auch eine genauere Erklärung des unten stehenden Beispiels.

Syntax:

```
<metaGroup>
  <meta Metafeldname>
    [Gruppdefinition(en)]
  </meta>
</metaGroup>
```

Beispiel:

```
<metaGroup>
  <meta charset>
    <groupname "CZECH">
      match = "ISO-8859-2"
    </groupname>
    <groupname "RU">
      match = "ISO-8859-5|KOI.?"
    </groupname>
    <groupname "INTL">
      match = "UTF-8"
    </groupname>
    <groupname "WESTERN">
      match = "ISO-8859-1"
    </groupname>
  </meta>
</metaGroup>
```

Meta-Gruppen lassen sich außerdem noch innerhalb von Directory-Blöcken einrichten.

<option>

Das Option-Tag beschreibt verschiedene Direktiven zur Steuerung der Indexerstellung. Die Beschreibung der einzelnen Direktiven finden Sie in Abschnitt 12.5.3.3.6 **Option-Direktiven** auf Seite 176.

Syntax:

```
<option>
  [Direktiven]
</option>
```

Beispiel:

```
<option>
  verbosityLevel = 1
  followSymLinks = ON
</option>
```

<directory>

Das Directory-Tag dient in erster Linie zur Festlegung der Verzeichnisse, die indiziert werden sollen. Außerdem können Sie es als Container für weitere Parameter nutzen. Die Indizierung des documentDir, die der DOCUMENT-ROOT Ihres Systems entspricht, ist per Default aktiviert.

Syntax:

```
<directory "pfad/zum/verzeichnis">
  [Direktiven]
</directory>
```

Absolute Pfadangaben notieren Sie mit einem führenden Slash bzw. einem Laufwerksbuchstaben gefolgt von einem Doppelpunkt. Alle anderen Pfadangaben werden relativ zum documentDir ausgewertet.



Tipp

Wenn Sie die Indizierung des documentDir deaktivieren wollen, notieren Sie folgende Directory-Definition:

```
<directory "">
  index = 0
</directory>
```

Beispiele:

Beim folgenden Beispiel handelt es sich um eine absolute Pfadangabe. Das Verzeichnis entspricht dabei dem Default-Wert für die Direktive documentDir (siehe Abschnitt 12.5.3.6 **Option-Direktiven** auf Seite 176). Alternativ könnten Sie auch mit einem Leerstring (<directory="">) darauf verweisen.

```
<directory /imperia/htdocs>
  [Direktiven]
</directory>
```

Das folgende Beispiel zeigt eine relative Pfadangabe. Das Verzeichnis FTS befindet sich direkt unterhalb des documentDir-Verzeichnisses.

```
<directory FTS>
  [Direktiven]
</directory>
```

Die Beschreibung der einzelnen Direktiven finden Sie in Abschnitt 12.5.3.5 **Tags und Direktiven in Directory-Blöcken** auf Seite 172.

<directoryMatch>

Wenn Sie statt eines Verzeichnisnamens Bestandteile des Verzeichnisnamens als Kriterium für die Anwendung bestimmter Direktiven einsetzen wollen, nutzen Sie das Tag <directoryMatch>. Im einleitenden Tag definieren Sie einen regulären Ausdruck. Dieser wird bei der Indexerstellung auf die einzelnen Verzeichnisse angewendet. Passt der reguläre Ausdruck auf den Verzeichnisnamen, kommen die im entsprechenden DirectoryMatch-Block definierten Direktiven zur Anwendung.



Hinweis:

Es werden nacheinander nur die einzelnen Verzeichnisnamen geprüft. Es ist nicht möglich, auf Bestandteile des gesamten Verzeichnispfades über Verzeichnistrenner hinweg zu prüfen.

Syntax:

```
<directoryMatch = "[regulärer_Ausdruck]">
  [Direktiven]
</directoryMatch>
```

Beispiel:

```
<directoryMatch = ".+print$">
  index = 0
</directoryMatch>
```

Nach dieser Definition würden folgende Verzeichnisse nicht indiziert:

- nachrichten_print
- produkte_print

Trotzdem indiziert würden jedoch folgende Verzeichnisse:

- nachrichten_print/doc
- nachrichten_print/pdf
- produkte_print/doc
- produkte_print/pdf
- print

Dieses Beispiel soll verdeutlichen, dass die Direktiven immer nur auf das Verzeichnis angewendet werden, dessen Name auf den regulären Ausdruck passt. Liegt ein passender Verzeichnisname im Pfad eines Verzeichnisses, haben die Direktiven dafür keine Gültigkeit.

<files>

Mit dem Files-Tag können Sie Direktiven für bestimmte Dateien festlegen. Im einleitenden Tag definieren Sie dabei mit einem regulären Ausdruck, welche Dateien beachtet werden. Die Direktiven werden auf Dateien angewendet, deren Name auf das Muster passt.

Syntax:

```
<files = "regulärer Ausdruck">
  [Direktiven]
</files>
```

Beispiele:

```
<files = "parse_this.html">
  index = 2
</files>
```

Die Datei parse_this.html wird indiziert.



Hinweis:

Die Direktive index=2 aktiviert die Indizierung. Der Wert 2 wird innerhalb von files-Blöcken wie 1 bzw. yes bzw. true gewertet. Lesen Sie hierzu auch Abschnitt 12.5.3.3.5 Tags und Direktiven in Directory-Blöcken auf Seite 172.

```
<files \.text>
  index = 1
  chmap = "UTF-8"
</files>
```

Dateien mit der Endung `.text` werden indiziert. Für Erläuterungen zu den in den Beispielen verwendeten Direktiven lesen Sie bitte Abschnitt 12.5.3.5 **Tags und Direktiven in Directory-Blöcken** auf Seite 172.

Dieses Tag können Sie auch in Directory- und DirectoryMatch-Blöcken einsetzen.

125333 Direktiven in Exclude-Blöcken

fileMetaKey

Bestimmen Sie ein Metafeld, bei dessen Vorhandensein das betreffende Dokument nicht indiziert werden soll. Der Wert dieser Direktive wird als regulärer Ausdruck gewertet.

Syntax:

```
fileMetaKey = "regulärer_Ausdruck"
```

metaKey

Geben Sie ein Metafeld an, dessen Inhalt nicht indiziert werden soll. Der Wert dieser Direktive wird als regulärer Ausdruck gewertet.

Syntax:

```
metaKey = "regulärer_Ausdruck"
```

Beispiel für einen Exclude-Block mit Direktiven:

```
<exclude>
  fileMetaKey "^mod.*"
  fileMetaKey "^layout_test$"
  metaKey "imperia"
</exclude>
```

Das Beispiel legt folgende Ausschlusskriterien fest:

- Dateien, die ein Metafeld enthalten, dessen Name mit "mod" beginnt, wobei dahinter weitere Zeichen folgen können, werden nicht indiziert.
- Dateien mit dem Metafeld `layout_test` werden nicht indiziert.
- Inhalte von Metafeldern, die im Namen die Zeichenfolge "imperia" enthalten, werden nicht in den Index aufgenommen.

125334 Direktiven in Include-Blöcken

fileMetaKey

Bestimmen Sie ein Metafeld, bei dessen Vorhandensein das betreffende Dokument indiziert werden soll. Der Wert dieser Direktive wird als regulärer Ausdruck gewertet.

Syntax:

```
fileMetaKey = "regulärer_Ausdruck"
```

Beispiel für einen Include-Block mit Direktiven:

```
<exclude>
  fileMetaKey "^mod.*"
  fileMetaKey "^layout_test$"
</exclude>
```

Das Beispiel definiert folgende Metafelder als Kriterium für ein Indizieren eines Dokuments:

- Dateien, die ein Metafeld enthalten, dessen Name mit "mod" beginnt, wobei dahinter weitere Zeichen folgen können, werden indiziert.
- Dateien mit dem Metafeld *layout_test* werden indiziert.

125335 Tags und Direktiven in Directory-Blöcken

<pageMeta>

Mit diesem Tag bestimmen Sie, welche Metataginhalte und Imperia-Metafeldinhalte in den Index aufgenommen werden und unter welchem Namen sie dort referenzierbar sind.

Syntax:

```
<pageMeta>
  Name_des_Imperia_Metafelds = meta_alias_im_Suchtemplate
  Name_des_HTML-Metatags = meta_alias_im_Suchtemplate
</pageMeta>
```

Beispiel:

In PDF-Dateien sind neben dem eigentlichen Inhalt auch einige Metainformationen gespeichert. Allerdings liegen diese als Textinformationen in einem Format vor, auf das Sie nicht ohne Weiteres zugreifen können. Die Direktive pageMeta ermöglicht Ihnen, gezielt die Metainformationen eines PDF-Dokuments zu indizieren, um Sie bei der Ausgabe eines Suchergebnisses verwenden zu können. Das Programm `pdfinfo` liefert folgende Metainformationen zu einem PDF-Dokument:

```
Title:      Microsoft Word - MyTitle
Author:     John Doe
Creator:    PScript5.dll Version 5.2.2
Producer:   Acrobat Distiller 6.0 (Windows)
CreationDate: Fri Jul 8 10:28:06 2005
ModDate:   Tue Aug 9 11:18:17 2005
Tagged:    no
Pages:     23
Encrypted: no
Page size: 595 x 842 pts (A4)
File size: 343800 bytes
Optimized: no
PDF version: 1.5
```

Um beispielsweise Titel, Erstellungs- und Änderungsdatum einer PDF-Datei in den Index der Volltextsuche aufzunehmen, definieren Sie folgenden PageMeta-Block:

```
<pagemeta>
  title = dc_title
  creationdate = page_time
  moddate = mod_date
</pagemeta>
```

Die indizierten Meta-Tag- bzw. Imperia-Metafeld-Inhalte lassen sich im Ergebnis-Template ausgeben. Enthält ein Dokument in der Trefferliste die indizierten Meta-Tags oder Metafelder, erscheint der jeweilige Inhalt an der entsprechenden Stelle in der Trefferliste.

Enne Suchanfrage nach den Metafeld- bzw. Meta-Tag-Inhalten liefert ebenfalls Treffer. Auch, wenn die Inhalte nicht zum sichtbaren Text des Dokuments gehören, wie das zum Beispiel bei Meta-Tags der Fall ist.

Einzelheiten zur Ausgabe von in PageMeta-Blöcken indizierten Daten finden Sie im Kapitel "**Templates für die Volltextsuche**" im Programmierhandbuch.

Das PageMeta-Tag können Sie auch innerhalb von DirectoryMatch- und Files-Blöcken einsetzen.

dirGroup

Die Direktive `dirGroup` ermöglicht es, Verzeichnisse bzw. Verzeichnisinhalte zu Gruppen zusammenzufassen. Ein Verzeichnis kann dabei mehreren Gruppen angehören. So definierte Gruppen lassen sich mit dem Parameter `GROUP` beim Aufruf des Suchskripts zur Filterung verwenden (Lesen Sie hierzu auch den Abschnitt Übergabeparameter und Variablen in Suchtemplates im Kapitel "**Templates für die Volltextsuche**" im Programmierhandbuch).

Syntax:

```
dirGroup = [Gruppenname (n) ]
```

Beispiel:

```
<directory ="produkte_de">
  index = yes
  dirGroup = german
</directory>
<directory ="news_de">
  index = yes
  dirGroup = german
</directory>
<directory = "produkte_en">
  index= yes
  dirGroup = english
</directory>
<directory = "produkte_en">
  index= yes
  dirGroup = english
</directory>
```

Die Verzeichnisse `produkte_de` und `news_de` gehören nach oben stehender Definition zur Gruppe `german`, die Verzeichnisse `produkte_en` und `news_en` zur Gruppe `english`. Das gilt ebenfalls für die jeweiligen Unterverzeichnisse. Beispielsweise gehören auch die Verzeichnisse `produkte_de/camping` und `produkte_de/haushalt` zur Gruppe `german`.

Mehrere Gruppen notieren Sie durch Leerzeichen voneinander getrennt.

Beispiel:

```
<directory ="produkte_de">
  index = yes
  dirGroup = sortiment german produkte
</directory>
```

Diese Direktive ist auch innerhalb von DirectoryMatch-Blöcken anwendbar. Beachten Sie dabei die besondere Funktionsweise von `directoryMatch` (siehe Abschnitt 12.5.3.3.2 **Tags in Domain-Blöcken** auf Seite 168).

Beispiel:

```
<directoryMatch = ".+_de">
  index = yes
  dirGroup = german
</directory>
<directoryMatch = ".+_en">
```

```
index= yes
dirGroup = english
</directory>
```

Alle Verzeichnisse, deren Name auf "_de" endet, gehören zur Gruppe german. Alle Verzeichnisse, deren Name auf "_en" endet, gehören zur Gruppe english.

Im Unterschied zum vorhergehenden Beispiel schließt das etwaige Unterverzeichnisse nicht mit ein. Die Verzeichnisse produkte_de/camping und produkte_de/haushalt müssten also produkte_de/camping_de und produkte_de/haushalt_de benannt sein, um ebenfalls zur Gruppe german gezählt zu werden.

index

Mit der Index-Direktive steuern Sie die Indexerstellung. Sie können einstellen, ob Verzeichnisse bzw. Dateien indiziert werden. Bei Verzeichnissen steuern Sie, ob Unterverzeichnisse ebenfalls indiziert werden sollen. Einstellungen aus übergeordneten Verzeichnissen lassen sich durch Verwendung der Direktive für Unterverzeichnisse (`directory`) oder Dateien (`files`) überschreiben.

Zur Aktivierung einer rekursiv ausgeführten Indizierung setzen Sie die folgenden Werte ein:

- yes
- true
- 1

Um ein Verzeichnis oder eine Gruppe von Dateien von der Indizierung auszuschließen, verwenden Sie folgende Werte:

- no
- false
- 0

Hinweis:



Innerhalb von Directory-Tags gilt „yes“ als Default-Wert, wenn Sie keine Index-Direktive setzen.

Verwenden Sie einen beliebigen anderen Wert, um nur das jeweilige Verzeichnis indizieren zu lassen. Unterverzeichnisse werden dann nicht durchsucht, es sei denn, Sie haben dies durch anderslautende Direktiven explizit aktiviert.

Beispiel:

```
index = yes-no-descend
```

Hinweis:



In Files-Blöcken gibt es diese Option nicht. Dort gilt jeder Wert, der die Option nicht explizit abschaltet, als Aktivierung.

Diese Direktive ist auch innerhalb von DirectoryMatch- und Files-Blöcken anwendbar.

Hinweis:



Innerhalb von Files-Blöcken bezieht sich die Index-Direktive ausschließlich auf die betreffenden Dateien, nicht auf die Verzeichnisse, in denen sich diese befinden.

chmap

Mit dieser Direktive können Sie einen Standard-Zeichensatz für Verzeichnisse festlegen. Bei der Indexerstellung wertet der Parser diese Angabe aus. Gibt es in den indizierten Dateien eine abweichende Angabe, z. B. durch ein entsprechendes Meta-Tag, hat diese Vorrang. Als mögliche Werte kommen alle vom Perlmodul *Locale::Recode* unterstützten Zeichensätze in Frage.

Syntax:

```
chmap = "Zeichensatz"
```

Beispiel:

```
chmap= "ISO-8859-1"
```

Diese Direktive können Sie auch in DirectoryMatch-Blöcken verwenden.

inherit

Diese Direktive ermöglicht es Ihnen, Einstellungen für ein Verzeichnis an dessen Unterverzeichnisse weiterzuvererben, auch wenn Sie das Verzeichnis selbst nicht bzw. nicht rekursiv indizieren lassen wollen. Einstellungen sind mit *inherit* nur über eine Verzeichnisebene vererbbar. In Unterverzeichnissen gesetzte, widersprüchliche Direktiven überschreiben vererbte Einstellungen.



Hinweis:

Die Einstellung für „index“ lässt sich nicht vererben.

Syntax:

```
inherit
```

Beispiel:

```
<directory = "news_de">
  index = "0"
  chmap = "ISO-8859-1"
  inherit
</directory>

<directory = "news_de/wirtschaft">
  index = "1"
</directory>

<directory = "news_de/politik">
  index = "1"
</directory>

<directory = "news_de/sport">
  index = "not-recursive"
  inherit
</directory>

<directory = "news_de/wetter">
  index = "1"
  chmap = "utf-8"
</directory>

<directory = "news_de/sport/fliegenfischen">
  index = "1"
</directory>
```

```
<directory = "news_de/sport/eisstockschiessen">
  index = "1"
</directory>
```

In diesem Beispiel soll das Verzeichnis news_de nicht indiziert werden. Allerdings wird festgelegt, dass der Inhalt ISO-8859-1-encodiert ist. Diese Einstellung soll auf Unterverzeichnisse weitervererbt werden, weswegen die Direktive inherit gesetzt wird. In den Unterverzeichnissen news_de/wirtschaft, news_de/politik und news_de/sport gilt deswegen auch die in der übergeordneten Ebene gesetzte chmap-Direktive.

Ausnahme ist das Verzeichnis news_de/wetter, da hier mit chmap = "utf-8" ein anderer Zeichensatz eingestellt wird. Da das Verzeichnis news_de/sport wegen der Einstellung index = "not-recursive" auch nicht rekursiv durchsucht wird, muss mit einem weiteren Einsatz von inherit dafür gesorgt werden, dass die beiden Verzeichnisse news_de/sport/fliegenfischen und news_de/sport/eisstockschiessen ebenfalls als ISO-8859-1-encodiert gelten. Ohne diese weitere Inherit-Direktive würde die Zeichensatzeinstellung nicht weitervererbt.

Inherit lässt sich auch in Exclude-, Files-, MetaGroup-, PageMetaComment- und PageMeta-Blöcken einsetzen, wenn diese innerhalb eines Directory-Blocks stehen.

125336 Option-Direktiven

cacheSize

Diese Direktive gibt die Anzahl der Begriffe an, die in einer Cache-Datei enthalten sind. Wenn Sie sie auf den Wert 0 setzen oder entfernen, wird kein Cache erstellt. Werte kleiner als 100 sind nicht sehr empfehlenswert, da dann sehr kleine Dateien generiert werden und so je nach Größe der Zuordnungseinheiten Speicherplatz auf der Festplatte verschwendet wird.

Syntax:

```
cacheSize = [Zahl]
```

Beispiel:

```
cacheSize = 150
```

compress

Mit dieser Direktive wählen Sie das Kompressions-Plug-In aus. Um eine Liste der verfügbaren Plug-Ins zu erhalten, rufen Sie das Skript site/bin/fts_index.pl mit dem Parameter -t in Ihrer Imperia-Installation auf.



Hinweis:

Ein Wechsel des Kompressions-Plug-Ins erfordert möglicherweise auch eine Anpassung der Einstellung cacheSize.

Der Default-Wert ist 0 (kein Kompressions-Plug-In).

cacheDir

Mit dieser Direktive geben Sie das Verzeichnis an, in dem der Datei-Cache der Volltextsuche gespeichert wird.

Syntax:

```
cacheDir = "/absoluter/pfad"
```

Beispiel:

```
cacheDir = "/usr/local/share/imperia/site/fts/index/mandant1/cache"
```



Wichtig:

Wenn Sie diese Variable setzen, müssen Sie auch darauf achten, dass das Verzeichnis nur von einem Index bzw. für eine Domain genutzt wird. Unter Umständen, zum Beispiel wenn mehrere Domains auf sich überschneidende Datenbestände zugreifen, kann es jedoch sogar wünschenswert sein, dass zumindest Teile des Caches von den Domains gemeinsam genutzt werden. Bitte wenden Sie sich im Zweifelsfall an den Imperia-Support.

Das System legt für jede konfigurierte Domain automatisch ein Unterverzeichnis an, in dem dann der eigentliche Cache gespeichert wird.



Wichtig:

Dies gilt nur, wenn Sie den Default-Wert nutzen.

Der Default-Wert ist `dataDir/cache`.

dataDir "/absoluter/pfad"

Tragen Sie hier den absoluten Pfad ein, unter dem der erzeugte Index abgelegt wird. Bei der Indexerstellung wird automatisch für jede konfigurierte Domain ein eigenes, nach der jeweiligen Domain benanntes Verzeichnis unterhalb dieser `dataDir` angelegt. Der eigentliche Index liegt in diesem Unterverzeichnis.

Syntax:

```
dataDir = "/absoluter/pfad"
```

Beispiel:

```
dataDir = "/usr/local/share/imperia/site/fts/index/mandant1"
```

Der Default-Wert ist das Verzeichnis `site/fts/index` Ihrer Imperia-Installation. Dieser Wert muss in der Konfiguration der Indexerstellung (Datei `index.conf`) und der Suche (Datei `fts.conf`) gleich sein.

dataTempDir

Mit dieser Direktive legen Sie das temporäre Arbeitsverzeichnis für die Indexerstellung fest. Tragen Sie eine absolute Pfadangabe als Wert ein. Die temporären Dateien der Indexerstellung werden für jede konfigurierte Domain in einem nach der Domain benannten Unterverzeichnis abgelegt.

Syntax:

```
dataTempDir = "/absoluter/pfad"
```

Beispiel:

```
dataTempDir = "/usr/local/share/imperia/site/fts/index/tmp"
```

Der Default-Wert ist `site/fts/index/tmp`.

documentDir

Definieren Sie einen absoluten Pfad zum Verzeichnis, das die zu indizierenden Dokumente enthält.

Syntax:

```
documentDir = "/absoluter/pfad"
```

Beispiel:

```
documentDir = "/usr/local/share/imperia/htdocs"
```

Der Default-Wert ist das Imperia-Document-Root.

followSubDir

Mit dieser Direktive können Sie die rekursive Indizierung aktivieren bzw. deaktivieren.

```
followSubDir = [ON|OFF]
```

Die Default-Einstellung ist *ON*.

followSymlink

Mit dieser Direktive regeln Sie die Behandlung von symbolischen Links.

Syntax:

```
followSymlink = [ON|OFF]
```

Per Default ist diese Option aktiviert.

maxDepth

Mit dieser Direktive steuern Sie die maximale Verzeichnistiefe, die bei der Indexerstellung berücksichtigt wird.

Syntax:

```
maxDepth = [Zahl]
```

Der Default-Wert ist 100 (Verzeichnisebenen).

memUseControl numerischer Wert

Setzen Sie hier einen numerischen Wert ein, der die maximale Größe der Speicherbereiche für die Indexerstellung festlegt. Mit dieser Direktive beeinflussen Sie die Performance und allgemeine Nutzung des Arbeitsspeichers.

Syntax:

```
memUseControl = [Zahl]
```

Der Default-Wert ist 6400000 (Bytes).

verbosityLevel

Mit dieser Direktive kontrollieren Sie die Menge der Informationen, die während der Indexerstellung ausgegeben werden.

Syntax:

```
verbosityLevel = [0-3]
```

Mögliche Werte:

Wert	Beschreibung
0	"Minimal", nur die jeweiligen Schritte der Indexerstellung werden angezeigt.
1	"Normal", jeweilige Schritte der Indexerstellung und indizierte Dateien sowie etwaige Fehler bei der Erstellung der Dateiliste oder beim Parsing werden ausgegeben.
2	"Ausführlich", zusätzlich zu den normalen Meldungen werden ausgelassene Verzeichnisse und Statistiken der SR-Dateien ausgegeben.
3	Wie "ausführlich", zusätzlich werden ausgelassene Dateien in den verarbeiteten Verzeichnissen und eine Liste der momentan indizierten Verzeichnisse ausgegeben.

Tabelle 12.2. Einstellungen der Direktive *verbosityLevel*

12.5.3.7 Tags und Direktiven in MetaGroup-Blöcken

<meta>

Dieses Tag nutzen Sie, um innerhalb von MetaGroup-Blöcken Gruppen für einzelne Metafelder festzulegen. Folgende Metafelder sind für jedes Dokument verfügbar:

Feld	Beschreibung
charset	Dies beschreibt den Zeichensatz, in dem das Dokument ursprünglich encodiert ist - nicht zu verwechseln mit dem in der Imperia-internen Datenhaltung verwendeten Zeichensatz (UTF-8).
filename	Dies ist der Dateiname des Dokuments.
title	Dies ist der Dokumententitel, es sei denn, dieses Feld wurde durch entsprechende Konfiguration explizit geleert.

Tabelle 12.3. allgemein verfügbare Metafelder



Hinweis:

Bei Dokumenten, die nicht mit Imperia erzeugt wurden, zum Beispiel außerhalb Imperias generierten PDF-Dateien oder Microsoft-Office Dokumenten, ist das Feld title leer.

Syntax:

```
<meta = "Metafeldname">
  [Gruppedefinition(en)]
</meta>
```

Gruppen lassen sich für jedes in einem Dokument verfügbare Metafeld definieren. Den Namen des betreffenden Feldes notieren Sie im einleitenden Tag eines Meta-Blocks. Anschließend können Sie für das Metafeld eine oder mehrere Gruppen definieren. Dabei legen Sie den Namen einer Gruppe mit dem Tag *groupname* fest.

<groupname>

Den Namen einer Meta-Gruppe legen Sie mit diesem Tag fest. Gruppennamen sind frei wählbar, dürfen aber keine Leerzeichen oder Sonderzeichen außer Binde- und Unterstrich enthalten. Geben Sie statt eines feststehenden Namens einen regulären Ausdruck an, um dynamisch benannte Gruppen zu erhalten. Lesen Sie hierzu auch Abschnitt 12.5.3.8 **Dynamisch benannte Meta-Gruppen** auf Seite 180. Innerhalb des Tags definieren Sie ein Suchmuster für Metafeldinhalte.

Syntax:

```
<groupname "String | regulärer Ausdruck">
  [Suchmuster]
</groupname>
```

Das Suchmuster definieren Sie mit der folgenden Direktive.

match

Legen Sie einen regulären Ausdruck fest, der während der Indexerstellung auf das betreffende Metafeld eines Dokuments angewendet wird. Passt das in dieser Direktive festgelegte Muster auf den Inhalt des betreffenden Metafeldes, gehört das Dokument zu der entsprechenden Gruppe.

Beispiel für eine Definition von Metagruppen:

```
<metaGroup>
<meta charset>
<groupname "CZECH">
  match = "ISO-8859-2"
</groupname>
<groupname "RU">
  match = "ISO-8859-5|KOI.+"
</groupname>
<groupname "INTL">
  match = "UTF-8"
</groupname>
<groupname "WESTERN">
  match = "ISO-8859-1"
</groupname>
</meta>
</metaGroup>
```

In diesem Beispiel werden für das Metafeld *charset* folgende vier Metagruppen definiert:

- *CZECH*, alle Dokumente, deren Charset "ISO-8859-2", bzw. "iso-8859-2" enthält
- *RU*, alle Dokumente, deren Charset "ISO-8859-5", bzw. "iso-8859-5" oder "KOI" bzw. "koi" gefolgt von beliebigen Zeichen enthält
- *INTL*, alle Dokumente, deren Charset "UTF-8", bzw. "utf-8" enthält
- *WESTERN*, alle Dokumente, deren Charset "ISO-8859-1", bzw. "iso-8859-1" enthält



Hinweis:

Bei der Auswertung regulärer Ausdrücke wird die Groß- und Kleinschreibung grundsätzlich nicht berücksichtigt. Lesen Sie hierzu auch Abschnitt 12.5.3.2 **Reguläre Ausdrücke in der Konfiguration** auf Seite 163.

12538 Dynamisch benannte Meta-Gruppen

Es ist auch möglich, Gruppennamen um einen dynamischen Bestandteil zu ergänzen. Dieser dynamische Bestandteil ergibt sich aus dem Metafeldinhalt, der die Gruppenzugehörigkeit bestimmt. So können Sie dynamische Gruppennamen zusammensetzen. Auf diese Weise definierte Gruppen sind eigenständig. Die Syntax zur Erzeugung dynamischer Gruppennamen basiert auf der Funktionsweise der regulären Ausdrücke in Perl:

```
<metaGroup>
<meta Metafeldname>
<groupname "Gruppenname_ ${1}">
  match = "(muster1)"
  match = "(muster2)"
</groupname>
</meta>
</metaGroup>
```

Damit die regulären Ausdrücke als Suffix an den Namen der Gruppe angehängt werden können, müssen Sie die Suchmuster in Klammern setzen (Gruppierung in regulären Ausdrücken). Dann kann der Zugriff auf die Gruppe im Suchmuster mit \${1} erfolgen.

Beispiel:

```
<metaGroup>
<meta title>
  <groupname "planet-${1}">
    match = "(weather)"
    match = "(forecast)"
    match = "THIS IS NOT WHAT YOU WANT"
  </groupname>
</meta>
</metaGroup>
```

Diese Definition legt anhand des Inhalts des Metafelds *title* drei Gruppen fest:

- Dokumente mit dem Wort "weather" im *title* gehören zur Gruppe "planet-weather".
- Dokumente mit dem Wort "forecast" im *title* gehören zur Gruppe "planet-forecast".
- Dokumente mit dem Satz "THIS IS NOT WHAT YOU WANT" im *title* gehören zur Gruppe "planet-".
Die dritte Gruppe ist zu Demonstrationszwecken falsch notiert und liefert nicht das (wahrscheinlich) erwünschte Ergebnis. Bei der Match-Direktive fehlen die Klammern, daher ergibt sich aus dem Suchmuster und dem feststehenden Teil des Namens kein dynamischer Gruppenname. Außerdem enthält das Muster Leerzeichen, und diese sind in Gruppennamen nicht erlaubt.

12.5.4 Die Konfiguration der Suche (fts.conf)

Mit den Einstellungen in der Datei *fts.conf* konfigurieren Sie, wie die Volltextsuche im Index nach eingegebenen Suchbegriffen sucht. Sollten Sie bereits eine ältere Version der Imperia-Volltextsuche eingesetzt haben, bietet Ihnen das Skript *fts_conf_convert.pl* aus dem Verzeichnis *site/bin* Ihrer Imperia-8-Installation die Möglichkeit, Ihre alte Suchkonfiguration in das Format der neuen Volltextsuche zu konvertieren. Die notwendigen Dateien werden dabei automatisch angelegt.

Eine Beispieldatei mit der notwendigen Minimalkonfiguration finden Sie im Verzeichnis *site/config* unter dem Namen *fts.conf.sample*. Sämtliche Parameter verfügen über Default-Werte, die auch in der Beispieldatei zum Einsatz kommen.



Hinweis:

Durch die Default-Werte können Sie die Suche auch ohne die Datei *fts.conf* bzw. mit einer leeren Datei nutzen. Allerdings sollten Sie dann vorher prüfen, ob die Default-Konfiguration für Ihre Zwecke genügt.

12.5.4.1 Basis-Tags

blacklist

Mit dieser Direktive geben Sie eine oder mehrere Stopwortlisten an, die für alle konfigurierten Domains der Volltextsuche gültig sein sollen. Die Suche ignoriert in einer Stopwortliste aufgeführte Wörter in Suchbegriffen. Damit die Volltextsuche die Stopwörter auch nicht indiziert, müssen Sie die Direktive auch in der Indexkonfiguration *site/config/index.conf* setzen.

Syntax:

```
blacklist = BEZEICHNER
```

Mit *BEZEICHNER* geben Sie eine Stopwortliste an, die Sie mit dem Skript *site/bin/fts_blacklist.pl* in das System eingetragen haben. Lesen Sie hierzu auch Abschnitt 12.5.3.7 **Stopwortlisten anlegen und verwalten** auf Seite 194.

Wenn Sie mehrere Stopwortlisten einsetzen möchten, verwenden Sie die Direktive mehrfach, für jede Stopwortliste einmal.

Diese Direktive können Sie auch innerhalb von Domain-Blöcken verwenden, dann gilt die Stopwortliste nur für die betreffende Domain.

<domain>

Die Einstellungen für die Suche innerhalb eines Index fassen Sie mit dem Tag `domain` in einem Block zusammen. In der Datei `fts.conf` lassen sich mehrere dieser Blöcke einsetzen. Sie können also mehrere Sets von unterschiedlichen Sucheinstellungen in einer Datei verwalten.

Diese Sets können sich sowohl auf einen einzigen als auch auf mehrere verschiedene Datenbestände beziehen. Jedes Set bzw. jede konfigurierte Domain benötigt ein eigenes Suchskript (siehe auch Abschnitt 12.5.3.6 **Erstellung von Suchskripten** auf Seite 194).

Alle weiteren Einstellungen werden innerhalb eines Domain-Blocks notiert.

Syntax:

```
<domain = "domain_name">
  [Direktiven]
</domain>
```

Der Default-Wert für den Domänennamen ist "`default`" bzw. ein Leerstring.

12542 Tags in Domain-Blöcken

<option>

Das Option-Tag fasst alle indexbezogenen Direktiven zu einem Block zusammen.

Syntax:

```
<option>
  [Direktiven]
</option>
```

Die Erklärung zu den Direktiven in Option-Blöcken und Beispiele finden Sie unter Abschnitt 12.5.3.4.3 **Direktiven in Option-Blöcken** auf Seite 185.

<lang>

Innerhalb eines Lang-Blocks definieren Sie sprachspezifische Synonyme für Bestandteile eingegebener Suchbegriffe. So lassen sich etwa die logischen Operatoren zur Verknüpfung mehrerer Suchwörter in unterschiedliche Sprachen übersetzen. Mit dem CGI-Parameter `lang` übergeben Sie beim Absenden von Suchanfragen die gewünschte Sprachversion, was dann bei der Verarbeitung der Anfrage die in der Suchkonfiguration hinterlegten Übersetzungen aktiviert.

Syntax:

```
<lang "Sprachkürzel">
  Wort = Übersetzung
</lang>
```

Beispiel:

```
<lang "DE">
  or = oder
  not = nicht
  and = und
</lang>
```

Dieses Beispiel zeigt die Ersetzung der logischen Operatoren `OR`, `NOT` und `AND` durch ihre deutschen Übersetzungen, wie sie auch in der Standard-Suchkonfiguration enthalten ist. Schicken Sie nun eine Suchanfrage mit dem CGI-Parameter `lang=de`, bzw. `lang=DE` ab, interpretiert die Volltextsuche die Wörter **oder**, **nicht**, **und** als logische Parameter, wenn sie im Suchbegriff vorkommen.



Hinweis:

Dies gilt auch, wenn Sie oder, nicht, und als Stopwörter konfiguriert haben.

<map>

Mit dem Tag `map` definieren Sie einen Block, in dem Sie festlegen, wie die Systempfade der gefundenen Dokumente in der Ergebnisliste angezeigt werden. Der Pfad zum jeweiligen Verzeichnis wird dann bei der Verlinkung in der Trefferliste durch den Wert ersetzt, den Sie vorgegeben haben.

Syntax:

```
<map "absoluter/Systempfad">
  to = "[URL]"
</map>
```

Beispiel:

```
<map "/usr/local/share/imperia/htdocs">
  to = "http://ihr-server.net"
</map>
```

Per Default wird das Verzeichnis, das Sie in der Datei `system.conf` Ihres Systems als DOCUMENT-ROOT angegeben haben, durch das ABS-DOC-ROOT ersetzt.



Hinweis:

Beim Mapping werden lange Pfadangaben vor kurzen verglichen.

<output>

Zur Steuerung der Ausgabe des Suchergebnisses stehen Ihnen eine Reihe von Direktiven zur Verfügung, mit denen Sie beispielsweise die Anzahl der Ergebnisse festlegen, die pro Bildschirmseite angezeigt werden. Diese Anweisungen notieren Sie in einem Block, den Sie durch das Tag „`output`“ kennzeichnen.

Syntax:

```
<output>
  [Direktiven]
</output>
```

In der Datei `site/config/fts.conf.sample` Ihrer Imperia-Installation finden Sie eine Beispielkonfiguration mit den Default-Werten. Die Direktiven für die Steuerung der Ausgabe sind in dem Abschnitt über Output-Direktiven erklärt.

Tags zur Template-Steuerung

Mit den Tags zur Template-Steuerung legen Sie fest, wo sich die Templates auf dem Zielsystem befinden, die von der Volltextsuche verwendet werden. Sie haben die Möglichkeit, innerhalb eines Tags mehrere Alternativen anzugeben, so dass ein schneller Wechsel zwischen unterschiedlichen Layoutvarianten möglich ist. Die Auswahl der Variante geschieht über einen Parameter, den Sie dem Suchskript mitgeben.

Wenn Sie jeweils nur eine Template-Variante benötigen, diese aber durch direkte Links erreichbar machen wollen, müssen sich die Templates unterhalb des DOCUMENT-ROOT des Zielsystems befinden, das durchsucht werden soll. Lesen Sie hierzu auch den Abschnitt zur Programmierung von Suchtemplates im Programmierhandbuch.

Syntax:

```
<[Template-Steuertag]>
  [Zahl] = "/pfad/zum/template"
</[Template-Steuertag]>
```

Die Zahl ist dabei der Schlüssel, über den das betreffende Template beim Aufruf des Suchskripts referenziert wird. Möglich sind absolute Pfadangaben oder gültige, zum Verzeichnis cgi-bin relative Pfade.

<standardHTMLTemplate>

Innerhalb dieses Tags geben Sie die Templates für einfache Suchanfragen an.

Beispiel:

```
<standardHTMLTemplate>
  1 = "/usr/local/share/imperia/htdocs/search/search-form.default.html"
</standardHTMLTemplate>
```

<advancedHTMLTemplate>

Templates für verfeinerte Suchanfragen notieren Sie innerhalb dieses Tags.

Beispiel:

```
<advancedHTMLTemplate>
  1 = "/imperia/site/fts/templates/search-form.default.html"
</advancedHTMLTemplate>
```

<resultHTMLPage>

Notieren Sie hier die Templates für die Darstellung der Suchergebnisse.

Beispiel:

```
<resultHTMLTemplate>
  1 = "/imperia/site/fts/templates/search-form.default.html"
</resultHTMLTemplate>
```

<templateDynamicGroups> und <template>

Wenn Sie für die Indexerstellung Meta-Gruppen definiert haben, können Sie im Suchtemplate eine Select-Box zur Verfügung stellen, die Imperia automatisch mit den vorhandenen Gruppen befüllt. Das Select-Feld im Suchtemplate müssen Sie hierfür mit dem Namen *GROUP* versehen und diesen in doppelte Anführungszeichen setzen. Mit dem Tag <templateDynamicGroups> bestimmen Sie, welche Gruppen mit welchem Namen in welchem Template automatisch als Optionen erscheinen sollen.

Syntax:

```
<templateDynamicGroups>
  <template [Template-Nummer]>
    Meta-Gruppenname = "Text für Option"
  </template>
</templateDynamicGroups>
```

Mit dem Tag <template> legen Sie fest, in welchem Suchtemplate die automatische Generierung der Optionen erfolgen soll. Ersetzen Sie den Platzhalter [Template-Nummer] durch die Nummer des entsprechenden Templates, die Sie bei den Template-Tags (siehe oben) angegeben haben. Als Meta-Gruppennamen tragen Sie den Namen ein, den Sie in der Indexkonfiguration verwendet haben. Der Text für die Optionen erscheint dann im Suchtemplate in der Selectbox. Dazu ein Beispiel:

Auszug aus der fts.conf:

```
<templateDynamicGroups>
  <template 1>
    newsFb = "Fußballnachrichten"
    newsHb = "Handballnachrichten"
    newsEh = "Eishockeynachrichten"
```

```
_all_ = "Treffer aus allen Gruppen"  
</template>  
</templateDynamicGroups>
```

Eine Besonderheit ist der Gruppenname `_all_`. Bei Auswahl dieser Option durchsucht Imperia alle vorhandenen Gruppen.

Im entsprechenden Suchtemplate ist dazu folgendes Select-Feld definiert:

```
<select name="GROUP" multiple="multiple">  
    <option>Dummy</option>  
</select>
```

Beim Aufruf des Suchtemplates über das Suchskript aus der cgi-bin ersetzt Imperia die Dummy-Option durch die Gruppen aus der Datei `fts.conf`. Im Suchtemplate erscheint also folgender Code:

```
<select name="GROUP" multiple="multiple">  
    <option value="newsFB">Fußballnachrichten</option>  
    <option value="newsHb">Handballnachrichten</option>  
    <option value="newsEh">Eishockeynachrichten</option>  
    <option value="_all_">Treffer aus allen Gruppen</option>  
</select>
```

12543 Direktiven in Option-Blöcken

Die folgenden Direktiven müssen Sie innerhalb eines Option-Blocks notieren.

verbosityLevel

Mit dieser Direktive steuern Sie den Detailgrad der Informationen, die am Schluss der Suche im Index ausgegeben werden. Der Default-Wert ist 1.



Hinweis:

Derzeit unterscheiden sich die ausgegebenen Meldungen nicht.

dataDir

Mit dieser Direktive legen Sie das Verzeichnis fest, in dem sich der Index befindet.



Wichtig:

Dieser Wert muss in der Konfiguration der Indexerstellung (Datei `index.conf`) und der Suche (Datei `fts.conf`) identisch sein.

Syntax:

```
dataDir = "/absoluter/pfad"
```

Das System legt automatisch für jede konfigurierte Domain ein Unterverzeichnis in dem Verzeichnis an, das Sie mit `dataDir` festgelegt haben.

Beispiel:

```
dataDir = "/usr/local/share/imperia/site/fts/index/mandant1"
```

Der Default-Wert ist das Verzeichnis `site/fts/index` Ihrer Imperia-Installation.

showContext

Die Anzahl der Wörter, die als Kontext zu einem Treffer angezeigt werden, bestimmen Sie mit dieser Direktive. Wenn Sie den Wert für diese Einstellung auf 0 setzen, deaktivieren Sie die Anzeige von Kontextbegriffen in der Trefferliste.

Syntax:

```
showContext = [Zahl]
```

Beispiel:

```
showContext = 100
```

Der Default-Wert dieser Einstellung ist 100.

percentPrecision

Mit dieser Direktiven bestimmen Sie die Anzahl der Nachkommastellen bei der Anzeige des Werts für die Templatevariable `PAGE_PERCENT`. Der Default-Wert ist 0.

phpEnable

Aktivieren Sie diese Direktive, wenn Sie PHP-Code in Ihrem Suchtemplate parsen lassen möchten, bevor es an den Browser geschickt wird.

Syntax:

```
phpEnable = [Wert]
```

Mögliche Werte zur Aktivierung des PHP-Parsings sind:

- yes
- true
- 1

Um das Feature zu deaktivieren, setzen Sie die Direktive auf einen der folgenden Werte:

- no
- false
- 0

Beispiel:

```
phpEnable = "yes"
```

Per Default ist die Auswertung von PHP-Code deaktiviert..

phpPath

Wenn Sie die Auswertung von PHP-Code im Suchtemplate aktiviert haben, müssen Sie mit dieser Direktive den Pfad zum PHP-Interpreter angeben, wenn sich dieser nicht im Umgebungspfad befindet.

Syntax:

```
phpPath = "absoluter/Pfad/zum/Interpreter/Binary"
```

Beispiel:

```
phpPath = "/usr/local/bin/php"
```

Der Default-Wert ist "php".

queryMode

Normalerweise ist aus Performancegründen in Suchbegriffen nur eine Wildcard möglich und bei der Verwendung von Wildcards dürfen Suchbegriffe nur aus einem Wort bestehen. Dieses Standard-Verhalten können Sie mit der Direktiven `queryMode` ändern. Außerdem lässt sich noch ein Suchmodus aktivieren, der die Suche nach Metagruppen ohne Angabe eines Suchworts erlaubt. Die folgende Tabelle erklärt die möglichen Einstellungen.

Parameter	Erklärung
<code>strict</code>	Innerhalb von Suchbegriffen ist nur eine Wildcard zulässig (Default).
<code>loose</code>	Ermöglicht die Verwendung mehrerer Wildcards in mehreren Suchbegriffen.
<code>extraloose</code>	Ermöglicht die Verwendung mehrerer Wildcards innerhalb von Suchwörtern.
<code>partial</code>	Die Volltextsuche behandelt Suchwörter ohne Wildcards automatisch so, als wären der erste und der letzte Buchstabe eine Wildcard.
<code>metaonly</code>	Ermöglicht die Suche nach Metagruppen ohne Angabe von Suchwörtern. Der Suchbegriff kann dann ausschließlich aus Metagruppen-Namen bestehen. Als Ergebnis erhalten Sie alle Dokumente aus den betreffenden Metagruppen.

Tabelle 12.4. Einstellungen für queryMode: verschiedene Suchmodi

minValidCharacters

Diese Direktive gibt an, wie viele Zeichen außer einer Wildcard in einem gültigen Treffer enthalten sein müssen. Der Default-Wert, 0, ermöglicht es auch, lediglich eine Wildcard als Suchbegriff anzugeben.

highlight

Bei aktiver Kontextanzeige lassen sich die gefundenen Suchwörter in dem angezeigten Text hervorheben. Dabei hebt die Volltextsuche normalerweise das ganze Wort hervor, auch wenn beispielsweise bei einer Suche mit Wildcard nur ein Teil des Wortes tatsächlich im Suchbegriff vorkommt. Um diese Teile gesondert hervorzuheben, setzen Sie die Direktive `highlight` auf den Wert `partial`. Die gesonderte Hervorhebung ist im Standard-Stylesheet für die Volltextsuche in eigenen CSS-Klassen definiert. Wenn Sie ein eigenes Stylesheet nutzen wollen brauchen Sie folgende Klassen für die Formatierung der Hervorhebungen:

- `highlight`
- `highlight_partial`
- `highlight_meta`

additionalUserVars

Mit dieser Direktive können Sie die Liste der CGI-Parameter für automatisch generierte, selbstreferenzierende Aufrufe der Suche erweitern. Diese zusätzlichen Parameter erscheinen dann beispielsweise in den Aufrufen des Suchskripts bei der automatisch erstellten Verlinkung der Trefferseiten.

Syntax:

```
additionalUserVars = "komma-separierte_Liste"
```

Beispiel:

```
additionalUserVars = "var1,var2,var3"
```

groupBool

In der Konfiguration der Indexerstellung können Sie mit den Direktiven `metaGroup` und `dirGroup` Gruppen definieren, die bei Suchanfragen als Filterkriterien fungieren, die mit dem Parameter `GROUP` an das Suchskript übergeben werden (Lesen Sie hierzu auch den Abschnitt „Übergabeparameter und Variablen in Suchtemplates“ im Kapitel „Templates für die Volltextsuche“ im Programmierhandbuch).

Mit der Direktive `groupBool` können Sie die boolesche Verknüpfung dieser Filtergruppen steuern. Mit der Einstellung `"AND"` aktivieren Sie eine Filterung über alle Gruppen. Jeder andere Wert bedeutet, dass Ergebnisse lediglich zu einer der vorhandenen Gruppen gehören müssen.

Syntax:

```
groupBool = [WERT]
```

Beispiele:

```
groupBool = "AND"
```

```
groupBool = "and"
```

Dies sind Beispiele für logische UND-Verknüpfungen. Treffer müssen zu allen mit `GROUP` übergebenen Gruppen gehören, um angezeigt zu werden.

```
groupBool = "OR"
```

```
groupBool = "any"
```

Dies sind Beispiele für logische ODER-Verknüpfungen. Treffer müssen nur einer der mit `GROUP` übergebenen Gruppen angehören, um in der Trefferliste zu erscheinen.

Diese Filterung ist nur aktiv, wenn Sie beim Aufruf des Skripts mehr als eine Gruppe als Filterkriterium mit dem Parameter `GROUP` übergeben. Der Default-Wert ist `"AND"`.

defaultOperator

Mit dieser Direktive geben Sie an, wie die Volltextsuche mehrere Wörter in Suchbegriffen verknüpft, wenn der Benutzer keine Operatoren eingibt. Der Standardwert ist `"OR"`.

resultFilterPlugin

Mit dieser Direktive aktivieren Sie auf Ihrem System vorhandene Ergebnisfilter zur Aufbereitung der Trefferliste vor der Ausgabe. Notieren Sie als Wert den Package-Namen des Filter-Plug-Ins. Ergebnisfilter-Plug-Ins müssen im Verzeichnis `site/modules/core/Dynamic/FTSResultFilter` liegen.

Syntax:

```
resultFilterPlugIn = Plug-In1,Plug-In2,Plug-InN
```

Mehrere Plug-Ins notieren Sie als kommaseparierte Liste ohne Leerzeichen. Dabei legen Sie gleichzeitig die Abarbeitungsreihenfolge der einzelnen Filter fest. Die Volltextsuche wendet den zuerst notierten Filter auch zuerst auf die Ergebnisliste an. Das Ergebnis dieser Filterung filtert sie anschließend mit dem an zweiter Stelle stehenden Filter und so weiter. Anschließend erfolgt die Ausgabe der gefilterten Ergebnisliste.

Beispiele:

Imperia beinhaltet drei verschiedene Ergebnisfilter:

DateBonus

Dieses Plug-In ermöglicht die Sortierung der Filter anhand datumsbasierter Bonuskriterien. Die Vergabe datumsbasierter Boni konfigurieren Sie mit einer Reihe weiterer Direktiven. Lesen Sie hierzu Abschnitt 12.5.3.4.7 **DateBonus: Treffer mit datumsbasierten Boni sortieren** auf Seite 192.

UserSort

Dieser Filter legt die Sortierreihenfolge der Trefferliste fest. Per Default erfolgt die Sortierung nach absteigender Relevanz. Alternative Sortiermethoden bestimmen Sie mit dem CGI-Parameter **SORT**, der im Abschnitt **Templates für die Imperia-Volltextsuche** im Programmierhandbuch beschrieben ist. Mit der Direktive **sortBy** bestimmen Sie die Default-Sortiermethode. Lesen Sie hierzu Abschnitt 12.5.3.4.6 **UserSort: Sortierung der Treffer bestimmen** auf Seite 191.



Hinweis:

Wie Imperia die Relevanz eines Dokuments berechnet, ist in Abschnitt 12.4 **Berechnung der Treffer-Relevanz und Boni** auf Seite 160 beschrieben.

XpdfConvert

Dieser Filter ist nützlich, wenn Sie PDF-Dokumente indizieren und auf der Ergebnisseite das Erstellungs- bzw. letzte Änderungsdatum der PDF-Dokumente anzeigen möchten. Das Programm **pdfinfo**, Bestandteil des XPDF-Pakets, liefert beide Datumsangaben als String im englischen Datumsformat. Dabei können abhängig von der zur Erzeugung des PDF-Dokuments verwendeten Software unterschiedliche Variationen auftreten.

Der Filter **XpdfConvert** wandelt ausgehend von einer Reihe von Varianten diese Datumsangaben in eine Angabe im Format **TT.MM.JJJJ hh:mm** um.

125344 Direktiven in parseroption-Blöcken

Parseroptions sind eine Unterebene der normalen Options. Ihre Besonderheit ist, dass sie direkt an alle Parser-Plug-Ins weitergereicht werden. Dadurch ist eine Erweiterung der Konfiguration für neu geschriebene Parser-Plug-Ins möglich.

Beispiel:

```
<option>
    dataDir = "/absoluter/pfad"
    <parseroption>
        ignoreMeta = "title|author"
    </parseroption>
</option>
```

ignoreMeta

ignoreMeta ist eine Option, die vom HTML-Parser ausgewertet wird. Mit ihrer Hilfe ist es möglich, den Inhalt von Metadaten aus dem HTML-Kopf zu ignorieren. Dies ist insbesondere wichtig, wenn wie im folgenden Listing ein Metatag "title" und ein HTML-Titel vorhanden ist.

```
<html>
    <head>
        <title>Seitentitel</title>
        <meta name="title" content="Seitentitel">
    </head>
```

Ohne Verwendung von **ignoreMeta="title"** würde der Titel hier zweimal indexiert und im Suchergebnis zweimal hintereinander ausgegeben.

Der Wert des Feldes wird als regulärer Ausdruck ausgewertet.

Beispiel:

```
ignoreMeta = "title|author"
```

keepphptag

keepphptag ist eine Option, die vom HTML-Parser ausgewertet wird. Ohne diese Option entfernt der HTML-Parser alle Tags, die mit <? ... ?> eingeschlossen sind, bevor der Rest der Datei geparsed wird.

Beispiel:

```
<option>
    <parseroption>
        keepphptag = 1
    </parseroption>
</option>
```

process_ssi

Wird die Option process_ssi auf "1" gesetzt, so werden beim Indexieren der Dokumente SSI-Includes mit eingebunden. Standardmäßig ist diese Option abgeschaltet, was dazu führt, dass Teile von Webseiten, die als SSI eingebunden sind, nicht mit indiziert werden.

Beispiel:

```
<option>
    <parseroption>
        parse_ssi = 1
    </parseroption>
</option>
```

12345 Direktiven in Output-Blöcken

CURRENT_PAGE

Mit dieser Direktive legen Sie fest, welche Ergebnisseite zuerst angezeigt wird.

Syntax:

```
CURRENT_PAGE = [Zahl]
```

Beispiel:

```
CURRENT_PAGE = 1
```

Der Default-Wert ist 1.

MAX_PAGE_LIST

Dies gibt die Höchstanzahl der Verweise auf weitere Trefferseiten an, die pro Seite angezeigt werden.

Syntax:

```
MAX_PAGE_LIST = [Zahl]
```

Beispiel:

```
MAX_PAGE_LIST = 100
```

Der Default-Wert ist 100.

MAX_PAGES

Dies gibt die Anzahl maximal generierter Trefferseiten an.

Syntax:

```
MAX_PAGES = [Zahl]
```

Beispiel:

```
MAX_PAGES = 100
```

Der Default-Wert ist 100.

PER_PAGE

Dies gibt die Anzahl angezeigter Treffer pro Bildschirmseite an.

Syntax:

```
PER_PAGE = [Zahl]
```

Beispiel:

```
PER_PAGE = 10
```

Der Default-Wert ist 10.

12.5.4.6 UserSort: Sortierung der Treffer bestimmen

Der Ergebnisfilter *UserSort* regelt die Sortierung der Trefferliste. Die jeweilige Sortiermethode legen Sie mit dem CGI-Parameter *SORT* fest, der im Abschnitt **Templates für die Imperia-Volltextsuche** im Programmierhandbuch beschrieben ist. Die Default-Sortiermethode bestimmen Sie mit der im Folgenden beschriebenen Direktive *sortBy*.

sortBy

Diese Direktive regelt die Default-Sortiermethode der Trefferliste. Die folgende Tabelle zeigt die verfügbaren Sortiermethoden:

Parameter	Erklärung
arelevance	aufsteigende Sortierung nach Relevanz
drelevance	absteigende Sortierung nach Relevanz (Default)
amodified	aufsteigende Sortierung nach Änderungsdatum
dmodified	absteigende Sortierung nach Änderungsdatum
asize	aufsteigende Sortierung nach Dateigröße
dsizes	absteigende Sortierung nach Dateigröße
awords	aufsteigende Sortierung nach Wortanzahl
dwords	absteigende Sortierung nach Wortanzahl

Tabelle 12.5. Sortiermethoden

disableUserSort

Setzen Sie diese Direktive auf den Wert 1, um etwaige mit der Suchanfrage übergebene Sortieroptionen zu unterdrücken.

12547 DateBonus: Treffer mit datumsbasierten Boni sortieren

Imperia bietet einen Ergebnisfilter, mit dem Sie Dokumenten anhand ihrer Aktualität einen Relevanzbonus verleihen können. Wenn Sie den Filter aktiviert haben, definieren Sie Maximalbonus, Intervall und Schrittweite der datumsbasierten Boni. Den Maximalbonus bekommt das jeweils aktuellste Dokument aus einer Trefferliste. Dieser Wert verringert sich im von Ihnen definierten Intervall mit abnehmender Aktualität des betreffenden Dokuments um die die ebenfalls von Ihnen vorgegebene Schrittweite.

bonus_date_max

Mit der Direktive `bonus_date_max` definieren Sie den Relevanzbonus für das aktuellste Dokument aus der Trefferliste. Geben Sie einen Wert zwischen 1 und 100 an.

Beispiel:

```
bonus_date_max = 50
```

bonus_date_step

Mit dieser Direktive geben Sie an, um wieviel sich der Bonus jeweils nach Ablauf eines Intervalls verringern soll. Möglich sind Werte zwischen 1 und 100.

```
bonus_date_step = 10
```

bonus_date_period

Diese Direktive gibt das Intervall an, in dem sich der Bonus für die Aktualität eines Dokuments verringert. Geben Sie eine Zahl und optional eine Zeiteinheit an. Für die Zeiteinheiten gibt es eine Kurz- und eine Langschreibweise, die in der folgenden Tabelle aufgeführt sind.

Kurzform	Langform	Beschreibung
<i>mi</i>	<i>minutes</i>	Minuten
<i>d</i>	<i>days</i>	Tage
<i>w</i>	<i>weeks</i>	Wochen
<i>mo</i>	<i>months</i>	Monate
<i>y</i>	<i>years</i>	Jahre

Tabelle 12.6. Zeiteinheiten für die Direktive `bonus_date_period`

Wenn Sie keine Zeiteinheit angeben, wählt die Volltextsuche automatisch Monate. Aus den drei Parametern Maximalbonus, Schrittweite und Zeitintervall ergibt sich, wie alt ein Dokument maximal sein kann, um einen Aktualitätsbonus zu erhalten. Dazu ein Beispiel:

```
bonus_date_max = 50
bonus_date_step = 10
bonus_date_period = 1 d
```

Mit dieser Konfiguration bekommen Dokumente, die bis zu einem Tag alt sind, einen Bonus von 50, was dem Maximalwert entspricht. Dokumente, die zwischen einem und zwei Tage alt sind, erhalten einen Bonus von 40 (50 - 10). Mit jedem weiteren Tag verringert sich der Bonus jeweils um weitere 10, bis er ganz aufgebraucht ist. Dokumente, die älter als 5 Tage sind, bekommen also gar keinen Bonus mehr.

bonus_date_field

Per Default nutzt die Volltextsuche das Metafeld `__imperia_modified`, das Datum der letzten Bearbeitung im CMS, zur Berechnung des Datumsbonus'. Grundsätzlich können Sie aber jedes Feld als Basis für diese Berechnung verwenden, das eine Datumsangabe in Form eines Unix-Timestamps enthält. Mit der Variablen `bonus_date_field` bestimmen Sie dieses Metafeld. Folgende Optionen stehen Ihnen zur Verfügung:

Parameter	Erklärung
file_modified	Datum der letzten Bearbeitung in Imperia (Default)
fs_file_modified	Änderungsdatum auf Dateisystemebene
METAFELDNAME	Beliebiges Metafeld, das einen Unix-Timestamp enthält. Definieren Sie in der Indexkonfiguration eine PageMeta-Direktive für dieses Feld, um es indizieren zu lassen.

Tabelle 12.7. Angaben für die Variable `bonus_date_field`

Wenn Sie ein beliebiges, selbst definiertes Metafeld als Grundlage für die Berechnung verwenden möchten, müssen Sie es durch die Definition einer entsprechenden PageMeta-Direktive in der Indexkonfigurationsdatei in den Index aufnehmen (siehe Abschnitt 12.5.3.3.5 **Tags und Direktiven in Directory-Blöcken** auf Seite 172).

Beachten Sie auch, dass das Änderungsdatum auf Dateisystemebene sich beispielsweise auch bei einem Reparsing der Datei ändert. Das bedeutet, dass nicht immer zwingend eine Änderung am Inhalt der Datei erfolgt, wenn sich das Datei-Änderungsdatum erneuert.

12.5.5 Indexerstellung

Die Erstellung des Suchindex stoßen Sie mit dem Skript `fts_index.pl` im Verzeichnis `site/bin` Ihrer Imperia-Installation an. Rufen Sie das Skript von der Kommandozeile aus auf. Dabei stehen folgende Aufrufparameter zur Verfügung:

Parameter	Bedeutung
<code>-h, --help</code>	Dieser Parameter gibt die Hilfeseite aus.
<code>-i, --install</code>	Die Installation eines zuvor erstellten Index wird ausgeführt. Die zugehörigen Dateien werden dabei aus dem temporären Verzeichnis in das mit der Einstellung <code>dataDir</code> spezifizierte Verzeichnis kopiert.
<code>-b, --build</code>	Löst die Erstellung des Index für die angegebenen Domains aus.
<code>-t, --test</code>	Startet die Prüfung der Index-Konfiguration und gibt eine Liste verfügbarer Komprimierungs-Plug-Ins aus.

Tabelle 12.8. Einstellungen der Direktive `verbosityLevel`

Außerdem können Sie beim Aufruf noch die Domains angeben, die Sie indizieren möchten. Jede Domain, die Sie hierbei aufführen, muss in der Datei `index.conf` im Verzeichnis `site/config` des Systems, das Sie indizieren möchten, wenigstens mit einem Domain-Block definiert sein.

Beispiel:

```
perl fts_index.pl -bi mandant1 mandant2 mandant3
```

Das Index-Skript liest die Konfigurationsdatei ein und erstellt für alle spezifizierten Domains, die definiert sind, einen Index. Wenn Sie beim Aufruf keine Domain angeben, wird die Standard-Domain "default" verwendet. Liegt im Verzeichnis `site/config` keine Konfigurationsdatei vor, setzt das Skript für alle Parameter automatisch die Default-Werte ein.

Der so erzeugte Index liegt zunächst in einem temporären Verzeichnis und ist erst verfügbar, wenn Sie das Skript zur Indexerstellung mit der Option `-i` bzw. `--install` ausführen.

Indexerstellung und `-install` können Sie auch in einem Arbeitsgang durchführen. Bei umfangreichen Datenbeständen kann die Indizierung erhebliche Zeit beanspruchen. Es ist empfehlenswert, den Index in regelmäßigen Abständen neu zu erstellen. Dies kann zeitgesteuert und automatisiert erfolgen, beispielsweise durch einen Cron-Job.

12.5.6 Erstellung von Suchskripten

Wenn eine Suchanfrage gestartet wird, liest das Suchskript die Einstellungen für das Durchsuchen des Index der zu durchsuchenden Domain aus der Datei `fts.conf` im Verzeichnis `site/config` ein. Wenn Sie mehrere Domains konfiguriert haben, brauchen Sie daher für jede Domain ein eigenes Suchskript.

Um ein domainspezifisches Suchskript zu erstellen, genügen wenige Schritte:

1. Erstellen Sie eine Kopie des Skripts `cgi-bin/fts_search.pl`.
2. Öffnen Sie die Datei in einem Editor und suchen Sie die folgende Zeile (Zeile 28):

```
my $domain = 'default';
```

3. Ändern Sie die Zeile wie folgt:

```
my $domain = '[Name_Ihrer_Domain]';
```

Beispiel:

```
my $domain = 'mandant1';
```

Speichern Sie die Datei.



Wichtig:

Das Installationsskript für Hotfixe benennt möglicherweise die im System vorhandene Version der Datei `cgi-bin/fts_search.pl` um und installiert die Ursprungsversion des Skripts. Daher sollten Sie domainspezifische Anpassungen immer in einer Kopie der Datei vornehmen und diese statt der Originaldatei für den Einsatz der Volltextsuche verwenden.

Bei Suchanfragen für diese Domain verweisen Sie nun auf diese bearbeitete Kopie des Suchskripts.

12.5.7 Stopwortlisten anlegen und verwalten

Um eine Stopwortliste für die Imperia-Volltextsuche einzurichten, sind folgende Schritte notwendig:

1. Stopwortlisten-Datei erstellen

Sie benötigen zunächst eine gewöhnliche ASCII-Textdatei mit beliebiger Dateiendung, die Sie an beliebiger Stelle in der Imperia-Verzeichnisstruktur ablegen.



Tipp

Es ist auch möglich, diese Datei als mit Imperia pflegbares Dokument zu verwalten. Beachten Sie in diesem Fall aber, dass die so erzeugte Textdatei im Bereich liegt, der über den Webserver erreichbar ist. Eventuell ist es also notwendig, den Zugriff auf diese Datei einzuschränken, beispielsweise durch einen Passwortschutz.

In der Datei sind die einzelnen Stopwörter jeweils in einer eigenen Zeile aufzuführen wie das folgende Beispiel zeigt:

```
ist
und
im
in
an
am
bei
beim
ein
```

```
eine
dem
der
die
das
```

Neben den eigentlichen Stopwörtern benötigen Sie in der Datei keinerlei weitere Anweisungen. Einmal angelegte Stopwortlisten können Sie jederzeit ersetzen, ändern oder entfernen.

2. Stopwortlisten-Eintrag generieren

Eine Stopwortlisten-Datei kann die Imperia-Volltextsuche noch nicht direkt verarbeiten. Um eine Stopwortlisten-Datei für das System verfügbar zu machen, rufen Sie das Skript `site/bin/fts_blacklist.pl` wie folgt auf:

```
perl fts_blacklist.pl -b BEZEICHNER pfad/zur/stopwortliste.end
```

Den mit dem Parameter `-b` spezifizierten *BEZEICHNER* verwenden Sie später in der Konfiguration der Indexerstellung und der Suche zur eigentlichen Aktivierung der Stopwortliste. Pfadangaben ohne führenden Slash interpretiert das System relativ zum Skript, mit führendem Slash absolut. Das Skript erzeugt in einem eigenen Verzeichnis parallel zum Suchindex eine mit dem *BEZEICHNER* benannte Datei, die mit dem Perl-Modul Storable serialisiert und somit für die Volltextsuche verarbeitbar ist.

Weitere Skriptparameter dienen zur Verwaltung bestehender Stopwortlisten. Die folgende Tabelle listet alle verfügbaren Parameter auf und erklärt sie.

Parameter	Bedeutung
<code>-h, --help</code>	Dieser Parameter gibt die Hilfeseite aus.
<code>-i, --input_charset</code>	Zeichensatz der Stopwortlisten-Datei, Default ist UTF-8.
<code>-b, --blacklist_name</code>	Pflichtparameter, der den Bezeichner der Stopwortliste angibt. Das System speichert die Wörter aus der angegebenen Stopwortlisten-Datei unter diesem Bezeichner, der auch zur Einbindung in der Konfiguration der Indexerstellung und Suche dient.
<code>-a, --append</code>	Das Skript hängt die Wörter aus der angegebenen Datei an die mit <code>-b</code> <i>BEZEICHNER</i> spezifizierte Stopwortliste an, statt diese zu ersetzen.
<code>-l, --list</code>	Listet alle im System verfügbaren Stopwortlisten-Einträge auf. Hierfür brauchen Sie weder Bezeichner noch Stopwortlisten-Datei anzugeben.
<code>-d, --delete</code>	Löscht den mit <code>-b</code> <i>BEZEICHNER</i> spezifizierten Stopwortlisten-Eintrag aus dem System. Eine Stopwortlisten-Datei brauchen Sie nicht anzugeben. Es ist nicht möglich, auch Stopwortlisten-Dateien mit diesem Parameter zu löschen.

Tabelle 12.9. Parameter für `site/bin/fts_blacklist.pl`

Dazu einige Beispiele:

```
perl fts_blacklist.pl -b mylist ../config/mylist.conf
```

Dieser Aufruf erzeugt aus der unter `site/config` liegenden Stopwortlisten-Datei `mylist.conf` einen Stopwortlisten-Eintrag, der unter dem Bezeichner `mylist` verfügbar ist. Einen etwaigen bereits bestehenden Stopwortlisten-Eintrag ersetzt diese Skriptausführung durch den neuen Eintrag.

```
perl fts_blacklist.pl -a -b mylist /home/wwwrun/imperia/stopwords/extra_words.txt
```

Dieser Aufruf hängt die Wörter aus der unter `/home/wwwrun/imperia/stopwords/` liegenden Stopwortlisten-Datei `extra_words.txt` an den Stopwortlisten-Eintrag an, der unter dem Bezeichner `mylist` verfügbar ist. Sollte noch kein Eintrag mit diesem Namen vorhanden sein, legt Imperia ihn an.

```
perl fts_blacklist.pl -d -b mylist
```

Dieser Aufruf löscht den Stopwortlisten-Eintrag, der unter dem Bezeichner `mylist` verfügbar ist.

3. Stopwortlisten in die Konfiguration einbinden

Stopwortlisten aktivieren Sie, indem Sie die Stopwortlisten-Einträge in den Konfigurationsdateien für Indexerstellung (`site/config/index.conf`) und Suche (`site/config/fts.conf`) angeben. Hierfür nutzen Sie in beiden Konfigurationsdateien die Direktive `blacklist`.

Syntax:

```
blacklist = BEZEICHNER
```

Der `BEZEICHNER` ist hierbei der Name des Stopwortlisten-Eintrags, den Sie beim Aufruf von `site/bin/fts_blacklist.pl` mit dem Parameter `-b` verwendet haben. Verwenden Sie die Direktive mehrmals, wenn Sie mehrere Stopwortlisten einsetzen wollen. Wenn Sie mehrere Volltextsuchen-Domains definiert haben, sind Stopwortlisten so definierbar, dass sie für alle oder nur für einzelne Domains gelten. Setzen Sie die Direktive `blacklist` innerhalb eines Domainblocks, damit die Stopwortliste nur für diese Domain gilt, und außerhalb der Domainblöcke in der Konfigurationsdatei, damit die Stopwortliste für alle Domains gültig ist.



Wichtig:

Wenn Sie Stopwortlisten in die Konfiguration der Indexerstellung einbinden, wirken diese sich erst bei der nächsten Indexerstellung aus. Das Gleiche gilt auch bei Änderungen an bereits bestehenden und eingebundenen Stopwortlisten.

Auf Suchanfragen wirken sich Stopwortlisten jedoch sofort nach ihrer Einbindung in die Suchkonfiguration aus. Die Suche ignoriert die Stopwörter sofort unabhängig davon, ob sie noch im Index sind oder nicht.

Kapitel 13. Anhang

13.1 Beispiele für LDAP.conf

Im Folgenden finden Sie jeweils ein Beispiel einer `LDAP.conf` für die beiden Gruppierungsschemata PAM und ADIR.

13.1.1 PAM

```
# Servername and IP address for LDAP server.  
serverldap.imperia.net  
port389  
  
# DN and password from ldap administrator  
# None if we are using the Anonymous Binding.  
#bind_dn "cn=Admin, o=intern"  
#password superuser  
  
# Container for user entries and attribute for RDN. So the DN of every  
# user have the form  
# "user_rdn_attribute=value_of_user_rdn_attribute, user_dn"  
user_dn_1 ou=People, o=Imperia, c=de  
#user_dn_2 ou=Verkauf, o=intern  
#user_dn_3 ou=Einkauf, o=intern  
#user_dn_4 ou=Vertrieb, o=intern  
#user_dn_5 "ou=Vertrieb, o=intern"  
user_rdn_attribute cn  
  
# Timelimit for search operations  
timelimit 0  
  
# Setting attributes for Imperia 8 user.  
USERFIELDS  
id uidnumber  
name cn  
login uid  
END_USERFIELDS  
valid_user (gidnumber=100)  
  
# Setting objectclasses an Imperia 8 user belongs to.  
#OBJECTCLASSES_USER  
#top  
#account  
#posixAccount  
#shadowAccount  
#person  
#organizationalPerson  
#inetOrgPerson  
#END_OBJECTCLASSES_USER  
  
# Setting attribute for Imperia 8 groups  
GROUPFIELDS  
id gidnumber  
name cn  
comment cn  
END_GROUPFIELDS  
  
#  
group_dn_1 ou=Groups, o=Imperia, c=de  
  
valid_group (gidnumber>=0)  
  
# Setting objectclasses an Imperia 8 group entry belongs to.  
#OBJECTCLASSES_GROUP  
#top
```

```
#posixGroup
#END_OBJECTCLASSES_GROUP

grp_schemePAM
superuseruid=tomg, ou=People, o=Imperia, c=de

13.1.2 ADIR

# Servername and IP adress for LDAP server.
serverwindb
port389

# DN and password from ldap administrator
# None if we are using the Anonymous Binding.
bind_dn Administrator@activetest.hyper
password xxxxxxxx

# Container for user entries and attribute for RDN.
user_dn_1 CN=Users,DC=activeTest,DC=hyper
user_rdn_attribute cn

# Timelimit for search operations
timelimit 0

# Setting attributes for Imperia 8 user.
USERFIELDS
id objectSid
name displayName
login cn
END_USERFIELDS

valid_user (objectclass=user)

# Setting objectclasses an Imperia 8 user belongs to.
#OBJECTCLASSES_USER
#END_OBJECTCLASSES_USER
# Setting attribute for Imperia 8 groups
GROUPFIELDS
id objectSid
name cn
comment description
END_GROUPFIELDS

#
group_dn_1 CN=Users,DC=activeTest,DC=hyper
group_dn_2 CN=Builtin,DC=activeTest,DC=hyper
valid_group (objectclass=group)

# Setting objectclasses an Imperia 8 group entry belongs to.
#OBJECTCLASSES_GROUP
#END_OBJECTCLASSES_GROUP

grp_schemeADIR
superuserCN=Administrator,CN=Users,DC=activeTest,DC=hyper
superuser_group CN=Domänen-Admins,CN=Users,DC=activeTest,DC=hyper
#userbind1
```

13.2 Dokumentenhaltung in einer Datenbank

13.2.1 Aufbau der Imperia-Tabellen

Folgende Tabellen werden in der Datenbank angelegt:

Tabelle	Funktion
archive_meta	Beinhaltet Metainformation der Dokumente, die sich im Archiv befinden. Entspricht dem Inhalt der Dateien unter site/documents/archive/*/*.arch/ bei Datenhaltung <i>File65</i> .
category	Beinhaltet alle Informationen zu Rubriken und Unterrubriken.
category_acl	Beinhaltet Zugriffsrechte der Rollen für Rubriken.
category_meta	Beinhaltet Metainformationen zu Rubriken. In dieser Tabelle liegt per Default die nodeID einer Rubrik. Zusätzlich in der Rubrik gespeicherte Metainformationen liegen ebenfalls in dieser Tabelle.
currentfiles	Beinhaltet Daten der Dokumente, die den Workflow bereits verlassen haben, aber noch nicht freigeschaltet wurden.
desktop	Beinhaltet Daten, die für die Darstellung des Schreibtischs benötigt werden.
document	Beinhaltet alle Informationen zu Dokumenten sowie Teile der früheren Meta-Datei.
document_acl	Beinhaltet Zugriffsrechte der Rollen für Dokumente.
history	Beinhaltet die Dokumenten-History.
image	Beinhaltet Binärdaten zu einem Dokument (Bilder, PDFs etc.).
longlive	Beinhaltet Daten der Dokumente, die bereits freigeschaltet wurden.
meta	Beinhaltet Metainformationen eines Dokuments, das sich im Workflow befindet.
object	Beinhaltet alle erstellten Imperia-Tabellen, Sequenzen und SQL-Funktionen.
sequence_table	Beinhaltet eine Tabelle mit Counter-Daten. Diese enthält die Zähler für Rubriken, Dokumente, History-Einträge und andere Tabellen der Datenhaltung.
ticket_data	Beinhaltet serialisierte Daten zum Workflow. Enthält außerdem die Daten, die in der Vorversion in den step-Tabellen gespeichert wurde.
version	Beinhaltet die Versionsnummern von Imperia, der RDBMS und des RDBMS-Treibers.

Tabelle 13.1. Datenbank-Tabellen für Imperia

13.2.1 Attribute in den Imperia-Tabellen

Im Folgenden finden Sie eine Beschreibung der einzelnen Attribute der Tabellen von Imperia.



Achtung:

Änderungen an Struktur oder Inhalt der Tabellen der Imperia-Datenhaltung können zu Datenverlusten oder Funktionsstörungen führen!

Die hier aufgeführten Angaben dienen zu rein informativen Zwecken.

13.2.1.1 Tabelle archive_meta

Spalte	Inhalt
doc_id	Fortlaufende Nummer der Dokumente wie in Tabelle document.
arch_id	Eindeutige Archivierungsnummer des Dokuments.
arch_flag	Enthält ein p, sobald das Dokument freigeschaltet wird.
meta	Metadaten-Block
meta_seq	Sequenznummer des Datenblocks.

Tabelle 13.2. Tabelle archive

13.2.1.2 Tabelle category

Spalte	Inhalt
cat_id	Rubrikinnummer
node_id	Rubriken- und Dokumenten-ID in neuer Form. Es wird eine HEX-Schreibweise anstelle der Slash-Ziffern-Schreibweise verwendet.
name	Name der Rubrik, der beim Anlegen der Rubrik angegeben wird.
parent_id	Nummer der übergeordneten Rubrik. Hat bei einer Hauptrubrik den Wert 0.
descr	Beschreibung der Rubrik, die beim Anlegen der Rubrik angegeben wurde.
metafile	Name der Metadatei.
workflow	Bezeichnung des Workflows.
template	Bezeichnung des Templates. Entspricht dem Inhalt der Klammer in einer Template-Auswahl.
directory	Der Pfad, in den das Dokument auf dem Zielsystem geschrieben werden soll.
time_stamp	Erzeugungszeitpunkt der Rubrik.
filename	Dateiname, der für alle Dokumente einer Rubrik verwendet werden soll. Wird beim Anlegen einer Rubrik angegeben; Standard ist index.html.
ownerid	ID des Users, der die Rubrik angelegt hat.
orderid	Ordnungsnummer für die Reihenfolge, in der Rubriken auf dem Bildschirm dargestellt werden.
nonewdocs	Flag (0 oder 1), das bestimmt, ob in dieser Rubrik Dokumente erstellt werden können oder nicht (Radio-Button beim Anlegen der Rubrik).

Tabelle 13.3. Tabelle category

13.2.1.3 Tabelle category_acl

Spalte	Inhalt
cat_id	Nummer der Rubrik bzw. Unterrubrik.
role_id	Rollen-ID (negative Zahl).
access_mode	Zugriffsrechte der Rolle (ein Datensatz pro Zugriffsrecht).

Tabelle 13.4. Tabelle category_acl

13.2.14 Tabelle category_meta

Spalte	Inhalt
cat_id	Nummer der Rubrik bzw. Unterrubrik.
meta_name	Name des Metafelds.
meta_value	Inhalt des Metafelds.
meta_idx	Index des Werts, falls das Metafeld ein Array enthält.
meta_seq	Sequenznummer des Datenblocks.

Tabelle 13.5. Tabelle category_meta

13.2.15 Tabelle currentfile

Spalte	Inhalt
record_id	Laufende Nummer für die interne Verwendung innerhalb von Imperia.
user_id	Benutzer-ID.
node_id	Komplette Rubriken- und Dokumenten-ID in neuer Form. Es wird eine HEX-Schreibweise anstelle der Slash-Ziffern-Schreibweise verwendet.
rec_date	Datum und Uhrzeit der Genehmigung des Dokuments.
url	URL (Pfad und Dateiname unterhalb DOC-ROOT) des Dokuments.
title	Titel des Dokuments.
publish_date	Datum des automatischen Freischaltens.
status	Momentaner Status des Dokuments.

Tabelle 13.6. Tabelle currentfile

13.2.1.6 Tabelle desktop

Spalte	Inhalt
doc_id	Fortlaufende Nummer der Dokumente wie in Tabelle <code>document</code> .
cat_id	Nummer der Rubrik.
node_id	Komplette Rubriken- und Dokumenten-ID in neuer Form. Es wird eine HEX-Schreibweise anstelle der Slash-Ziffern-Schreibweise verwendet.
directory	Pfad, unter dem das Dokument auf dem Zielsystem gespeichert werden soll.
filename	Dateiname, der für alle Dokumente einer Rubrik verwendet werden soll. Wird beim Anlegen einer Rubrik angegeben; Standard ist <code>index.html</code> .
template	Bezeichnung des Templates. Entspricht dem Inhalt der Klammer in einer Template-Auswahl.
metafile	Name der Metadatei.
title	Titel des Dokuments.
modified	Zeitstempel der letzten Modifikation.
created	Zeitstempel der Erstellung des Dokuments.
user_id	ID des Users, der das Dokument auf dem Schreibtisch sehen kann.
grid_status	Status des Dokuments im Workflow
locks	Information, ob das Dokument gesperrt ist.
publish_date	Datum, an dem das Dokument freigeschaltet werden soll.
expiry_date	Datum, an dem das Dokument vom Zielsystem automatisch gelöscht werden soll.

Tabelle 13.7. Tabelle desktop

13.2.1.7 Tabelle document

Spalte	Inhalt
doc_id	Fortlaufende Nummer der Dokumente.
cat_id	Nummer der Rubrik des Dokuments (Relation zur Tabelle <code>category</code>).
is_archived	Flag, ob das Dokument bereits archiviert ist.
node_id	Komplette Rubriken- und Dokumenten-ID in neuer Form. Es wird eine HEX-Schreibweise anstelle der Slash-Ziffern-Schreibweise verwendet.

Tabelle 13.8. Tabelle document

13.2.1.8 Tabelle document_acl

Spalte	Inhalt
doc_id	Fortlaufende Nummer der Dokumente wie in Tabelle <code>document</code> .
role_id	Rollen-ID, für die Zugriffsrechte gelten.
access_mode	Zugriffsrechte der Rolle (ein Datensatz pro Zugriffsrecht).

Tabelle 13.9. Tabelle document_acl

13.2.19 Tabelle history

Spalte	Inhalt
hist_id	Laufende Nummer der Einträge.
doc_id	Fortlaufende Nummer der Dokumente wie in Tabelle <code>document</code> .
time_stamp	Zeitstempel. Bei jeder Bearbeitung eines Dokuments wird ein neuer Datensatz mit aktuellem Zeitstempel geschrieben.
user_id	User-ID (Achtung: nicht mit der Rolle verwechseln!).
hist_data	Systemeinträge und Userbeschreibung des jeweiligen Arbeitsschritts.

Tabelle 13.10. Tabelle history

13.2.10 Tabelle image



Hinweis:

Diese Tabelle enthält rein binäre Daten und wird nicht näher beschrieben.

13.2.11 Tabelle longlive

Spalte	Inhalt
record_id	Laufende Nummer für die interne Verwendung innerhalb von Imperia.
user_id	Benutzer-ID
node_id	Komplette Rubriken- und Dokumenten-ID in neuer Form. Es wird eine HEX-Schreibweise anstelle der Slash-Ziffern-Schreibweise verwendet.
rec_date	Datum und Uhrzeit der Genehmigung des Dokuments.
url	URL (Pfad und Dateiname unterhalb <code>DOC-ROOT</code>) des Dokuments.
title	Titel des Dokuments.
publish_date	Datum des automatischen Freischaltens.
status	Momentaner Status des Dokuments.

Tabelle 13.11. Tabelle longlive



Hinweis:

Die Tabellen `currentfiles` und `longlive` sind identisch aufgebaut. Die Tabelle `currentfiles` enthält die Dokumente, die genehmigt sind und den Workflow verlassen haben, bevor sie freigeschaltet werden. Diese Tabelle wird ausgelesen, wenn der Menüpunkt **Dokument freischalten** benutzt wird.

Die Tabelle `longlive` enthält Dokumente, die auf dem Zielsystem existieren. Sie wird ausgelesen, wenn der Menüpunkt **Freigeschaltete Dokumente bearbeiten** verwendet wird.

13.11.2 Tabelle meta

Spalte	Inhalt
doc_id	Fortlaufende Nummer der Dokumente wie in Tabelle document.
meta_name	Meta-Variable (z.B.: filename).
meta_value	Wert der Meta-Variable (z.B.: index.html).
meta_idx	Index des Werts, falls das Metafeld ein Array enthält.
meta_seq	Sequence-ID der einzelnen Metainfo-Blöcke

Tabelle 13.12. Tabelle meta

13.11.3 Tabelle object

priority	Löscherienfolge der einzelnen Typen. Zum Beispiel Sequenzen vor Tabellen. Sichert den reibungslosen Ablauf eines Löschvorgangs.
type	Typ des Eintrags. Momentan relevante Werte: TABLE, SEQUENCE, FUNCTION
name	Name der Tabelle, Sequenz oder Funktion.

Tabelle 13.13. Tabelle object

13.11.4 Tabelle sequence_table

Spalte	Inhalt
name	Name des Counters.
start	Startwert des Counters.
minvalue	Minimalwert des Counters.
maxvalue	Maximalwert des Counters.
increment	Erhöhungssintervall des Counters.
cycle	Schalter für Reset des Counters bei Erreichen des Maximalwerts.
curval	Aktueller Wert des Counters.

Tabelle 13.14. Tabelle sequence_table

13.11.5 Tabelle ticket_data

Spalte	Inhalt
doc_id	Fortlaufende Nummer der Dokumente wie in Tabelle document.
ticket_data	Serialisierte Ticketdaten.
ticket_sequence	Sequenz-Nummer des Datenblocks.

Tabelle 13.15. Tabelle ticket_data

13.11.6 Tabelle version

Spalte	Inhalt
imperia_version	Version von Imperia. Wird bei der automatischen Erzeugung der Tabellen durch Imperia eingetragen.
db_version	Version der benutzten Datenbank.
db_driver	Informationen über den installierten DBD-Treiber.

Tabelle 13.16. Tabelle version

13.3 Die Imperia Admin-Skripte

Die Imperia-Admin-Skripte sind als Werkzeuge für Systemadministratoren gedacht. Sie können damit häufig auszuführende Aufgaben vereinfachen. Zur Ausführung rufen Sie das betreffende Skript von der Kommandozeile aus auf. Sie liegen im Verzeichnis `site/bin` Ihrer Imperia-Installation. Ein Zugriff über die Imperia-Benutzeroberfläche ist nicht möglich.



Wichtig:

Legen Sie vor der Verwendung der Admin Skripte ein Backup an bzw. stellen Sie sicher, dass ein aktuelles Backup existiert.

Versetzen Sie Imperia außerdem in den Wartungsmodus mit Superuserzugriff (siehe Abschnitt 3.2.2 Wartungsmodus mit Superuser-Zugriff auf Seite 18) bzw. stellen Sie sicher, dass kein anderer Benutzer über die Weboberfläche arbeitet.

Die Skripte liegen alle im Verzeichnis `site/bin`. Eine Erklärung zu ihrer Anwendung und zu den jeweils verfügbaren Optionen erhalten Sie mit dem Aufrufparameter `--help`. Zum Beispiel:

```
admin_accesscat.pl --help
```

13.3.1 admin_accesscat.pl

Mit diesem Skript können Sie die Zugriffsrechte für eine Rubrik anzeigen lassen und ändern.

13.3.2 admin_accessmeta.pl

Legen Sie die Zugriffsrechte für ein Metafile fest oder lesen Sie die gegenwärtigen Zugriffsberechtigungen.

13.3.3 admin_accesstemplate.pl

Mit diesem Skript ändern Sie die Zugriffsrechte für ein Template oder lesen die aktuelle Konfiguration aus.

13.3.4 admin_accessworkflow.pl

Mit diesem Skript ändern oder lesen Sie die Zugriffsrechte für einen Workflow.

13.3.5 admin_acl_update.pl

Mit diesem Skript löschen Sie alte, nicht mehr gültige Rollen aus den ACLs.

13.3.6 admin_addcat.pl

Mit diesem Skript legen Sie eine neue Rubrik oder Unterrubrik an.

Die Eigenschaften der neuen Rubrik definieren Sie mit Hilfe der Aufrufparameter. Dabei sind alle Angaben optional. Einzige Ausnahme ist der Rubrikname, den Sie angeben müssen.

Bei Unterrubriken müssen Sie zusätzlich den eindeutigen Namen oder die ID der übergeordneten Rubrik angeben.



Hinweis:

Nach der Verwendung von admin_addcat.pl sollten Sie die Rubrikenrechte mit dem Skript admin_accesscat.pl einrichten.

13.3.7 admin_adddoc.pl

Mit diesem Skript erstellen Sie ein neues Dokument, dessen Attribute Sie über die Aufrufparameter festlegen. Per Default wird anschließend der Workflow für das neue Dokument angestoßen. Dies können Sie jedoch auch durch einen Parameter unterbinden.

13.3.8 admin_addrole.pl

Mit diesem Skript legen Sie eine neue Rolle an. Den Namen der neuen Rolle und einen Kommentar können Sie beim Aufruf des Skriptes übergeben.

Mit *admin_deriverole.pl* (siehe auch Abschnitt 13.3.14 **admin_deriverole.pl** auf Seite 206) können Sie die neue Rolle anschließend einer anderen Rolle als Unterrolle zuteilen bzw. der neuen Rolle eine andere Rolle als Unterrolle zuweisen.

Nutzen Sie das Skript *admin_grpuser.pl* (siehe Abschnitt 13.3.21 **admin_grpuser.pl** auf Seite 207), um Benutzern Zugriff auf diese Rolle zu gewähren.

Für die Festlegung der Berechtigungen der neuen Rolle stehen Ihnen die Admin-Skripte *admin_accesscat.pl* (Rubrikenrechte, siehe auch Abschnitt 13.3.1 **admin_accesscat.pl** auf Seite 205), *admin_accesstemplate.pl* (Templates, siehe auch Abschnitt 13.3.3 **admin_accesstemplate.pl** auf Seite 205), *admin_accessmeta.pl* (Metafiles, siehe auch Abschnitt 13.3.2 **admin_accessmeta.pl** auf Seite 205) und *admin_accessworkflow.pl* (Workflows, siehe auch Abschnitt 13.3.4 **admin_accessworkflow.pl** auf Seite 205) zur Verfügung.

13.3.9 admin_adduser.pl

Nutzen Sie dieses Skript, um einen neuen Benutzer anzulegen. Die Attribute des neuen Benutzers können Sie beim Aufruf des Skriptes übergeben.

Zugriffsrechte erteilen Sie dem neuen Benutzer, indem Sie mit dem Skript *admin_grpuser.pl* seine Rollenzugehörigkeiten festlegen.

Sie können anderen Benutzern mit dem Skript *admin_edituser.pl* Zugriff auf den neuen Benutzer gewähren.

13.3.10 admin_cleandesktop.pl

Dieses Skript findet und löscht (shreddert) alle MAM Dokumente im Schreibtisch, die keine assoziierten Binärdaten enthalten. Dokumente dieser Art entstehen beim Abbruch des Uploads.

13.3.11 admin_currentfile.pl

Mit diesem Skript können Sie Ihre Site nach Dateien durchsuchen, die weder im longlive.log noch in current.files aufgeführt sind. Wahlweise können Sie sich eine Liste dieser Dateien anzeigen lassen oder diese direkt in den Datenbestand des CMS integrieren.

13.3.12 admin_delrole.pl

Mit diesem Skript löschen Sie eine Rolle aus dem System. Als Parameter können Sie die ID oder den eindeutigen Namen der Rolle angeben.

13.3.13 admin_deluser.pl

Mit diesem Skript löschen Sie einen Benutzer aus dem System. Als Parameter können Sie die ID oder den eindeutigen Login-Namen des Benutzers angeben.

13.3.14 admin_deriverole.pl

Nutzen Sie dieses Skript, um sich die bestehende Rollenhierarchie anzeigen zu lassen oder Veränderungen daran vorzunehmen. Sie können damit eine bestehende Rolle einer oder mehreren anderen als Unterrolle zuweisen.

13.3.15 admin_expire.pl

Mit diesem Skript können Sie Dokumente, die ein Verfallsdatum eingetragen haben, wahlweise anzeigen oder in die Datei site/articles/expiry.new schreiben lassen.

13.3.16 admin_getcat.pl

Mit diesem Skript können Sie die Eigenschaften einer Rubrik ausgeben. Es stehen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können. Die Rubrik legen Sie durch Angabe der NodeID (beginnend mit einem Slash) oder des eindeutigen Rubriknamens fest.

13.3.17 admin_getdoc.pl

Sie können sich mit diesem Skript die Metainformation zu einem Dokument ausgeben lassen. Dabei stehen Ihnen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können. Das Dokument bestimmen Sie durch die Angabe der NodeID beim Aufruf des Skripts.

Wenn Sie die Informationen einer archivierten Version des Dokumentes anzeigen lassen möchten, geben Sie zusätzlich die arch_id der betreffenden Version an. Andernfalls werden die Informationen der neuesten Version ausgegeben.

13.3.18 admin_getrole.pl

Mit diesem Skript können Sie die Eigenschaften einer Rolle anzeigen lassen. Die Rolle bestimmen Sie durch die Angabe der RollenID oder des eindeutigen Rollennamens. Es stehen Ihnen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können.

13.3.19 admin_getuser.pl

Mit diesem Skript können Sie die Eigenschaften eines Benutzers anzeigen lassen. Den Benutzer bestimmen Sie durch die Angabe der UserID oder des eindeutigen Login-Namens des Benutzers. Es stehen Ihnen verschiedene Ausgabeformate zur Verfügung, die Sie mit Aufrufparametern auswählen können.

13.3.20 admin_grpaccess.pl

Mit diesem Skript können Sie die Zugriffsberechtigungen auf die Rollen Ihres Systems anzeigen lassen oder ändern. Entziehen oder gewähren Sie Benutzern Zugriff auf eine Rolle.

13.3.21 admin_grpuser.pl

Mit diesem Skript können Sie sich anzeigen lassen, welche Rollen Ihres Systems ein Benutzer annehmen darf oder einem Benutzer Zugriff auf weitere Rollen gewähren.

13.3.22 admin_listcat.pl

Nutzen Sie dieses Skript, um sich eine Liste aller Rubriken unterhalb einer Startrubrik anzeigen zu lassen.

13.3.23 admin_listflex.pl

Rufen Sie dieses Skript auf, um sich eine Liste aller Flexmodule inklusive ihrer Validcodes, Invalidcodes und Autoren anzeigen zu lassen.

13.3.24 admin_listmeta.pl

Für eine Liste aller Metafiles Ihres Systems rufen Sie dieses Skript auf.

13.3.25 admin_listrole.pl

Mit diesem Skript können Sie sich eine Liste aller Rollen ausgeben lassen.

13.3.26 admin_listtemplate.pl

Mit diesem Skript können Sie eine Liste aller Templates erzeugen und ausgeben lassen.

13.3.27 admin_listuser.pl

Nutzen Sie dieses Skript, um sich eine Liste aller Benutzer anzeigen zu lassen.

13.3.28 admin_listworkflow.pl

Mit diesem Skript können Sie eine Liste aller Workflows anzeigen lassen.

13.3.29 admin_sanitisearchive.pl

Nutzen Sie dieses Skript für die Bereinigung des gesamten Archives oder einzelner Teilbereiche.

Geben Sie beim Aufruf eine NodeID oder einen eindeutigen Rubrikennamen an, um nur einen Teilbereich des Archivs zu bearbeiten. Sollen auch Unterrubriken bereinigt werden, müssen Sie dies beim Aufruf des Skripts vermerken.

Alternativ können Sie das ganze Archiv bereinigen lassen.

Für die Bereinigung stehen Ihnen zwei Optionen zur Auswahl. Sie können wahlweise nur sämtliche unbedienten Dokument-Versionen löschen, die älter sind als die jeweils letzte beendete, oder auch zusätzlich die Anzahl verbleibender beendeter Versionen einschränken.

13.30 admin_searchdoc.pl

Mit diesem Skript suchen Sie Dokumente nach bestimmten Suchkriterien.

Sie erhalten eine Liste von Dokumenten, auf die Ihre Suchkriterien zutreffen. Neben der NodeID können Sie sich eine frei definierbare Reihe von Metafeldinhalten der gefundenen Dokumente anzeigen lassen.

Zur Eingrenzung der Suche können Sie beim Aufruf die NodeID oder den eindeutigen Namen einer Startrubrik übergeben.

13.31 admin_searchuser.pl

Mit diesem Skript können Sie anhand bestimmter Suchkriterien nach Benutzern suchen.

Sie erhalten eine Liste aller Benutzer, auf die diese Kriterien zutreffen. Per Default enthält diese Liste die UID und den Login-Namen der gefundenen Benutzer. Weitere Felder können Sie durch Parameter beim Aufruf der Liste hinzufügen.

13.32 admin_setcat.pl

Mithilfe dieses Skripts können Sie die Eigenschaften bestehender Rubriken ändern.

Die Rubrik bestimmen Sie durch die Angabe einer NodeID oder eines eindeutigen Namens beim Aufruf.

Die Werte, die Sie ändern möchten, übergeben Sie in Form einer durch Leerzeichen getrennten Reihe von Name-Wertpaaren.

13.33 admin_setcounter.pl

Nutzen Sie dieses Skript, um die internen Counter von Imperia zu verändern. Sie haben Zugriff auf die Counter für Rubriken und Dokumente sowie für Verzeichnisse.

13.34 admin_setdoc.pl

Mit diesem Skript können Sie die Metainformationen eines Dokumentes verändern. Das Dokument wählen Sie durch die Angabe einer NodeID beim Aufruf aus.

Per Default werden die Informationen der letzten Version des Dokumentes verändert. Optional können Sie auch die der letzten beendeten Version oder die einer per archID bestimmten Version aus dem Archiv verändern.

13.35 admin_setrole.pl

Für die Änderung von Rolleneigenschaften können Sie dieses Skript nutzen.

Übergeben Sie dem Skript die ID oder den eindeutigen Namen der Rolle, die Sie ändern möchten, und eine durch Leerzeichen getrennte Reihe von Name-Wertpaaren mit den neuen Eigenschaften.

13.36 admin_setuser.pl

Mit diesem Skript können Sie die Eigenschaften eines Benutzers ändern.

Übergeben Sie dem Skript die ID oder den eindeutigen Namen des Benutzers, den Sie ändern möchten, und eine durch Leerzeichen getrennte Reihe von Name-Wertpaaren mit den neuen Eigenschaften.

13.37 admin_statdoc.pl

Mit diesem Skript können Sie Informationen zum Status eines Dokuments erhalten.

13.4 Liste aller Variablen zur System-Konfiguration

Imperia verfügt über eine Vielzahl von Möglichkeiten zur Systemkonfiguration. Diese Konfiguration geschieht über Variablen, die in Dateien abgespeichert werden.

Die für den Betrieb wichtigsten Variablen werden während des Installationsprozesses automatisch angelegt. Sie können sie über die Benutzeroberfläche in entsprechenden Dialogfenstern nachträglich ändern, beispielsweise die Einstellungen der Mediendatenbank oder die Konfiguration des Zielsystems.

Zusätzlich zu diesen grundlegenden Variablen gibt es noch eine Reihe weiterer Einstellungsmöglichkeiten, die es Ihnen erlauben, Ihr System an Ihre individuellen Anforderungen anzupassen.

Sie können sämtliche Konfigurationsdateien über die Benutzeroberfläche editieren. Lesen Sie hierzu Abschnitt 3.11 **Konfiguration des Systems über die Benutzeroberfläche** auf Seite 42.

Im Folgenden sind die verfügbaren Konfigurations-Variablen thematisch gruppiert aufgeführt.

13.4.1 Gruppe ACLs

13.4.1.1 Variable ENABLE_FLEX_ACL

Auf „1“ setzen, um ACLs für Flexmodule zu aktivieren.

13.4.2 Gruppe Anzeige

13.4.2.1 Variable BOTTOMTABLE

Geben Sie hier den Pfad zu einer Datei mit einem HTML-Schnipsel an, um die Footer-Tabelle in der Imperia-GUI auszutauschen. Diese Datei muss sich unterhalb des Document-Root des Produktionssystems befinden, z.B.: /my_cool_bottomtable.html.

13.4.2.2 Variable CATEGORY_DELETION_PROMPT

Wenn diese Variable den Wert „1“ hat, wird vor dem Löschen einer leeren Rubrik ein Bestätigungsdialog angezeigt.

13.4.2.3 Variable DISABLE-DATE-TOOLTIP

Bei allen Datumsanzeigen in Imperia wird im Tooltip die Serverzeit und die Userzeit angezeigt. Dies behindert in manchen Browsern bei vielen Datumsfeldern das flüssige Arbeiten. Durch setzen der Variablen „DISABLE-DATE-TOOLTIP“ auf „1“ werden diese Tooltips deaktiviert.

Diese Variable wurde mit Imperia Version 8.5 eingeführt.

13.4.2.4 Variable DISABLE-FAV-MENU

Mit dieser Variablen können Sie das Favoriten-Menü zu- oder abschalten. Setzen Sie sie auf „1“, um das Menü abzuschalten.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13.4.2.5 Variable DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER

Wenn Dokumente auf verschiedene Liveserver publiziert werden, können für das Veröffentlichen Bedingungen angegeben werden. Nicht erfüllte Bedingungen werden jeweils beim Freischalten von Dokumenten mit gelben Kästchen vermerkt. Wird die Variable DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER auf "1" gesetzt, werden keine Warnungen für nicht erfüllte Freischaltbedingungen mehr ausgegeben.

```
DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER = "1"
```

13.4.2.6 Variable DISABLE-STREAMED-OUTPUT

Gestreamte Datenübertragung abschalten. Es werden keine NPH-Scripte verwendet.

13.4.7 Variable DISABLE_TOOLTIP

Unterdrückt die Abkürzung von Dateipfaden in der Imperia-Oberfläche. Standardmäßig werden lange Dateipfade in Imperia abgekürzt. Dieses Verhalten kann mit der System-Conf-Variable unterdrückt werden. Als Werte müssen die Skripte angegeben werden, für welche das Verhalten geändert werden soll.

Beispiele:

```
DISABLE_TOOLTIP = "__site_ac:title,__site_ac:directory"  
DISABLE_TOOLTIP = "__site_free:title,__site_free:url,__site_mydesk:filename,__site_fromlive:url"
```

13.4.8 Variable DISPLAY_BASE_CLASS

Ist diese Variable gesetzt, wird für die Darstellung der Imperia-Oberfläche nicht das zur Installation gehörende Modul `site/modules/core/Imperia/Display/Base.pm` benutzt, sondern es kann ein projektspezifisch angepasste Display-Modul genutzt werden.

13.4.9 Variable DONT_ASK_BEFORE_EXPLODE_ALL_NODES

Wenn Sie diese Variable auf „1“ setzen, wird keine Warnmeldung angezeigt, wenn alle Rubriken des Rubrikenbaums angezeigt werden.

13.4.10 Variable FAV_MENU_LINK_ONLY

Durch diese Variable können Sie die Anzeige von Icons im Favoritenmenü zu- oder abschalten. Der Defaultwert ist 1 für abgeschaltete Icons.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13.4.11 Variable FROMLIVE_DELETE_DEFAULT_LIVE

Durch das Setzen dieser Variable auf den Wert "1" wird die Vorauswahl des Dropdownfeldes in der Freischaltliste auf den Wert "Dateien nur auf Zielsystem löschen" gesetzt.

13.4.12 Variable LINKTOOL_DISPLAY_ASSETS

Wenn die Variable auf den Wert "1" gesetzt wird, zeigt das Linktool sowohl normale als auch Asset-Rubriken an.

13.4.13 Variable GRID_HEIGHT

Mit dieser Einstellung legen Sie die vertikale Anzahl von Grid-Zellen im Workflow-Editor fest.

13.4.14 Variable GRID_WIDTH

Mit dieser Einstellung legen Sie die horizontale Anzahl von Grid-Zellen im Workflow-Editor fest.

13.4.15 Variable IBS_PAGE

Setzen Sie diese Variable auf einen Wert ≥ 0 , um im Archiv eine Paginierung zu aktivieren. Der Wert gibt dabei an, wie viele Dokumente auf einer Seite im Archiv erscheinen sollen. Die Anzeige von Rubriken und Dokumentversionen im Archiv beeinflusst diese Variable nicht.

13.4.16 Variable IMPERIA_SKIN_PATH

Pfad zu den Skinning-fähigen Komponenten.

Der Default-Wert ist `/imperia/skin`.

13.4.17 Variable LATCHING

Mit dieser Variablen schalten Sie die Funktion **Zwischenspeichern** für den Edit-Modus von Dokumenten ein. Im Bearbeiten-Schritt eines Dokuments erscheint dann bei den Steuerelementen ein zusätzlicher Button, mit dem sich eingegebener Inhalt speichern lässt, ohne den Workflow-Schritt zu verlassen.

134218 Variable MAIN_MENU_LINK_ONLY

Setzen Sie diese Variable auf 1, um die Anzeige der Icons im Hauptmenü zu unterbinden. Diese Einstellung gilt für alle Benutzer.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

134219 Variable MD-WINDOW-HEIGHT

Mit dieser Variablen können Sie die Höhe aller Mediendatenbank-Fenster festlegen. Die Angabe erfolgt numerisch und wird als Höhe in Pixeln umgesetzt.

134220 Variable MD-WINDOW-WIDTH

Mit dieser Variablen können Sie die Breite aller Mediendatenbank-Fenster festlegen. Die Angabe erfolgt numerisch und wird als Breite in Pixeln umgesetzt.

134221 Variable NO-NPH

Gestreamte Datenübertragung abschalten.

134222 Variable POPUP_WINDOW_HEIGHT

Mit dieser Variablen legen Sie die Höhe aller von Imperia geöffneten Pop-Up-Fenster fest. Die Angabe erfolgt numerisch und wird als Höhe in Pixeln umgesetzt.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

134223 Variable POPUP_WINDOW_WIDTH

Mit dieser Variablen legen Sie die Breite aller von Imperia geöffneten Pop-Up-Fenster fest. Die Angabe erfolgt numerisch und wird als Breite in Pixeln umgesetzt.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

134224 Variable TOPTABLE

Geben Sie hier den Pfad zu einer Datei mit einem HTML-Schnipsel an, um die Header-Tabelle in der Imperia-GUI auszutauschen. Diese Datei muss sich unterhalb des Document-Root des Produktionssystems befinden, z.B: /my_cool_toptable.html.

13.4.3 Gruppe Authentifizierung

13431 Variable AUTH_CHAINED_PLUGINS

Mit dieser Variablen definieren Sie eine Reihe von Authentifizierungs-Plug-Ins, die das Plug-In Dynamic::Auth::Chained nutzen.

13432 Variable AUTH_PLUGIN

Imperia Authentifizierungs-Plug-In.

13433 Variable AUTH_PLUGIN_BACKUP

In dieser Variablen wird der Authentifizierungsmechanismus aus AUTH_PLUGIN gespeichert wenn das System in den Maintenancemode umgeschaltet wird. Beim Zurückschalten wird dann dieser Wert wieder in die Variable AUTH_PLUGIN geschrieben.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13434 Variable COMMWORLD_AUTH_CHECKSESSION

Sessionstring, den das Auth-Plug-In des CommWorld/PortalBuilders nutzt.

13.4.4 Gruppe Datenhaltungs-Plug-In Datei

13.4.4.1 Variable FILE65-DOCUMENT-DIR

Verzeichnis, in dem das File65-Plug-In Dokumente abspeichert.

Der Default-Wert ist *SITE-DIR/documents*.

13.4.5 Gruppe Dynamische Variablen

13.4.5.1 Variable ABS-CGI-BIN

In dieser Variablen ist der absolute Pfad zum cgi-bin-Verzeichnis Ihres Webservers gespeichert. Alle CGI-Skripte von Imperia liegen in diesem Verzeichnis.



Hinweis:

Der Wert für diese Variable wird zur Laufzeit vom System generiert. Änderungen, die Sie von Hand daran vornehmen, wirken sich nicht aus.

13.4.5.2 Variable ABS-DOC-ROOT

Der absolute Pfad zum Stammverzeichnis für Webdokumente des Servers.



Hinweis:

Der Wert für diese Variable wird zur Laufzeit vom System generiert. Änderungen, die Sie von Hand daran vornehmen, wirken sich nicht aus.

13.4.6 Gruppe Exporter

13.4.6.1 Variable XALAN_CLASSPATH

Classpath für Xalan 2.1

13.4.6.2 Variable XSLTPROC_PATH

Pfad zum XSLT-Prozessor, der vom Exporter verwendet wird.

13.4.6.3 Variable XSLT_PLUGIN

Plug-In für den XSLT-Export. Derzeit werden folgende Plug-Ins unterstützt: XMLXSLT, Xalan, XSLTProc und LibXSLT.

13.4.7 Gruppe File Transfer

13.4.7.1 Variable DISABLE_WRITE_TRANSMISSION_CACHE

Wenn Sie den Transmission-Cache deaktivieren, werden alle Dateien jedesmal freigeschaltet, wenn Sie verwendet werden.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13.4.7.2 Variable OPEN_3_USE_SYSREAD

Diese Variable wird auf Linux-Systemen mit neueren Kernelversionen benötigt, wenn Sie Dateitransfer per Secure Copy einsetzen wollen. Setzen Sie sie auf 1, um Sysread bei Secure Copy Transfers zu aktivieren.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13.4.7.3 Variable SSH-OPTIONS

Über diese Variable können Optionen die bei der SSH-Verbindung verwendet werden sollen angegeben werden. Damit sich shell-Verbindungen und Imperia-Verbindungen gleich verhalten, sollten die Optionen besser in der *.ssh* Datei gepflegt werden.

13474 Variable SSH-CLIENT

Der absolute Pfad zur SSH-Binärdatei. Sie benötigen diese Variable, wenn der Default-Wert nicht zutrifft. Der Default-Wert ist *ssh*.

13475 Variable SSH-FILED

In dieser Variablen ist gespeichert, welches Filed-Skript auf Ihrem Zielsystem aufgerufen wird. Der Default ist *SITE-DIR/bin/filed.pl*. Dabei wird SITE-DIR durch die jeweils aktuelle Einstellung auf Ihrem Ziel-System ersetzt.

Der Default-Wert ist *SITE-DIR/bin/filed.pl*.

13.4.8 Gruppe Flexmodule

13481 Variable ALLOW_PERL_UPLOAD

Mit dieser Variable wird gesteuert, ob Perl-Flexmodule in Imperia bearbeitet werden können.

13482 Variable FLEXMODULE-SKIN

Mit dieser Variable geben Sie den Dateinamen für Ihr Flexmodul-Skin an, mit dem Sie systemweit das tabellenbasierte Standard-Layout der Flexmodul-Steuerleiste überschreiben möchten. Geben Sie nur den Dateinamen an, wenn Sie die Datei im Standard-Verzeichnis für Flexmodul-Skins, *site/skins/flex* ablegen möchten. Alternativ können Sie eine Pfadangabe ausgehend von der Document-Root Ihres Entwicklungssystems notieren. Für die Ersetzung des Codes der Flexmodul-Steuerleiste auf Dokumentenbasis verwenden Sie den Parameter *SKIN* im Flexmodul-Aufruf im Template.

13483 Variable FLEXINSTANCE-SKIN

Wie **FLEXMODULE-SKIN**, nur für die Steuerleiste der Flex-Instanzen.

13.4.9 Gruppe Grundeinstellungen

13491 Variable CGI-BIN

Diese Variable enthält den URI des cgi-bin-Verzeichnisses auf dem Webserver, das heißt den Pfad ausgehend vom Document-Root Ihres Servers, beispielsweise '*/cgi-bin*', '*/cgi-shell*', '*/cgi/perl*' usw.

Der Default-Wert ist */cgi-bin*.

13492 Variable CGI-DIR

In dieser Variablen ist der absolute Pfad zum cgi-bin-Verzeichnis Ihres Webservers gespeichert. Alle CGI-Skripte von Imperia liegen in diesem Verzeichnis.

13493 Variable DEMOMODE

Diese Variable können Sie als Hintertür zum System für Notfälle nutzen. Sollten Sie einmal keinen Zugriff mehr auf Ihr System haben, kommentieren Sie die entsprechende Zeile in der Datei *site/config/system.conf* aus. Sie können sich danach wieder am System anmelden und haben Superuser-Zugriffsrechte (Jeder Benutzer, der sich dann am System anmeldet, hat Superuser-Zugriffsrechte). Anschließend können Sie einen neuen Benutzer anlegen. Entfernen Sie den Kommentar vor der Zeile, um zum Normalbetrieb zurückzukehren.

Der Default-Wert ist *no*.

13494 Variable DISABLE-FLOCK

Setzen Sie diese Variable auf 1, um File Locking zu deaktivieren. Dies ist notwendig, wenn das Betriebssystem, unter dem Sie Imperia installieren, diesen Mechanismus nicht unterstützt (z.B. Windows 98).

13495 Variable DISABLE-FORTUNE-COOKIE

Setzen Sie diese Variable auf 1, um die Anzeige eines Fortune Cookies bei der Anmeldung generell für alle Benutzer zu unterbinden.

Der Default-Wert ist 1.

13496 Variable DOCTREE_PLUGIN

Verwendetes Datenhaltungs-Plug-In.

Der Default-Wert ist *File65*.

13497 Variable COUNTER-DIR

Pfad zum Counter-Verzeichnis.

13498 Variable DEBUG_ArchiveCleaner

Debug-Flag für das ArchiveCleaner Plug-In.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13499 Variable ARCHIVESHREDDER

Setzen Sie diese Variable auf 1, damit das System bei einem Löschtorgang auf dem Entwicklungs-System bzw. im Archiv automatisch auch die zugehörigen Dateien vom Zielsystem entfernt.

13500 Variable ARCHIVE_DISABLE_DELETE

Setzen Sie diese Variable auf "1", um die Löschfunktion im Archiv zu deaktivieren.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13501 Variable DISGUISE_JAVA_PLUGIN

Setzen Sie diese Variable auf 1, um die Bibliothek Imperia::HTML::Disguise zum Verbergen von Java Plugins und anderen ActiveX-Aufrufen zu nutzen.

13502 Variable FLOCK_DEBUG

Wenn Sie diese Variable auf „1“ setzen, aktivieren Sie das Debugging für Flock.

13503 Variable FollowSymLinks

Mit dieser Variablen können Sie Teile des Dateisystems einbinden, die außerhalb des Document-Root Ihres Systems liegen. Wenden Sie diese Variable nur in dringenden Fällen an, da Sie hiermit möglicherweise Ihr System für Angriffe verwundbar machen.

13504 Variable IMPERIA-VERSION

Imperia Versionsnummer, bitte nicht ändern.

13505 Variable IMPERIA_ERROR_LOG

Setzen Sie diese Variable auf „1“ für ein eigenes Error-Log. Die Datei finden Sie unter „`SITE-DIR/logs/error.log`“.



Hinweis:

Diese Variable wird in Windows-Installationen automatisch aktiviert.

13506 Variable MAX_REQUEST_SIZE

Mit dieser Variable legen Sie die maximale Größe in Bytes für Dateien beim Upload fest. Die Angabe muss numerisch erfolgen. Versuche, Dateien hochzuladen, die größer sind als dieser Wert, beantwortet das System mit einer Fehlermeldung. Wenn diese Variable nicht gesetzt wird, können Dateien von unbegrenzter Größe hochgeladen werden.

Diese Variable wurde mit Imperia Version 6.5.2 eingeführt.

134917 Variable MIME_TYPES

Mit dieser Variablen können Sie einen alternativen Pfad zu Ihrer MIME-Type-Datenbank definieren. Hier können projektspezifische Erweiterungen updatesicher eingetragen werden; beachten Sie, dass die benötigten Typen aus der Originaldatei mit eingetragen werden müssen, um in Kraft zu bleiben.

134918 Variable USE_STRICT

Hiermit erzwingen Sie für Perl-Codeincludes global die Einhaltung der Strict-Perl-Konventionen.

134919 Variable PACK_STR

Pack-String für Publisher::StrangeFile::Record.

134920 Variable SECRET_FREQUENCY

Anzahl der Sekunden, nach deren Verstreichen das Secret-Cookie erneuert wird.

Der Default-Wert ist 3600.

134921 Variable SECURE_UMASK

Maske zur Generierung des Secure-Verzeichnisses.

134922 Variable UMASK

Setzen Sie diese Variable, um Voreinstellungen Ihres Providers zu überschreiben.

134923 Variable DOCUMENT-ROOT

Diese Variable enthält den absoluten Pfad zum Document-Root Ihrer Imperia-Installation. Beispiel: „DOCUMENT-ROOT“ = „/usr/httpd/htdocs“ .



Hinweis:

Das Verzeichnis „imperia“ sollte sich unterhalb des Basis-Verzeichnisses für Webdokumente Ihres Webservers befinden, beispielsweise „/usr/http/htdocs/imperia“.

134924 Variable OPERATING-SYSTEM

Das Betriebssystem, unter dem Imperia installiert ist. Entweder „UNIX“ oder „DOS“, je nachdem, was dem von Ihnen verwendeten System "am nächsten" kommt.

Der Default-Wert ist UNIX.

134925 Variable SITE-DIR

Dies ist eine der wenigen Konfigurationsvariablen die nicht in der Datei /site/config/system.conf steht sondern in der Datei /cgi-bin/imperia.conf die unter anderem dem Auffinden der Datei /site/config/system.conf dient.

In dieser Variablen steht der absolute Pfad zum site-Verzeichnis Ihrer Installation.



Wichtig:

Aus Sicherheitsgründen sollten die drei Verzeichnisse SITE-DIR, CGI-BIN und DOCUMENT-ROOT voneinander getrennt sein. Das heißt, dass keines dieser drei Verzeichnisse sich innerhalb eines der anderen befinden sollte. Andernfalls sind unbeabsichtigte oder böswillige Zugriffe durch User über das Internet möglich.

13410 Gruppe HTTP Cookie

134101 Variable COOKIE-DOMAIN

Domain des Cookies.



Wichtig:

Bitte beachten Sie, dass der Internet Explorer keine Cookies akzeptiert, bei denen der Hostname Unterstriche "_" enthält.

134.102 Variable COOKIE-PATH

Cookie-Pfad.

134.103 Variable COOKIE_NAME

Name des Imperia-Cookies.

134.104 Variable COOKIE_TTL

Zur Bestimmung der Gültigkeitsdauer (Time To Live) des Imperia Cookies können Sie diese Variable verwenden. Die Angabe erfolgt numerisch und gibt die Lebensdauer des Cookies in Sekunden an. Ist ein Benutzer über den mit dieser Variable festgelegten Zeitraum untätig, wird seine Session automatisch gelöscht und er muss sich erneut anmelden. Der Default-Wert ist 7200 Sekunden.

134.11 Gruppe Hermes

134.11.1 Variable HERMES_RECONNECT

Durch setzen des Schalters HERMES_RECONNECT auf „1“ wird bei jedem Hermes durchlauf ein neuer Connect zur Datenbank aufgebaut.

Diese Variable wurde mit Imperia Version 8.5.4 eingeführt.

134.11.2 Variable MAX_AUTO_PUBLISH_DOCS

Diese Variable gibt die Höchstzahl von Dokumenten an, die innerhalb einer Schleife des Hermes-Daemons freigeschaltet werden. Der Default-Wert ist „25“.

134.11.3 Variable HERMES_INCOMING_DELAY

Bei einigen FTP-Servern ist der Lock-Mechanismus während Dateiübertragungen fehlerhaft. Mit dieser Variablen können Sie eine Anzahl von Sekunden angeben, die das MessageCheck-Plug-In mit der Verarbeitung nach der letzten Änderung einer Datei pausiert, um diesem Fehler entgegenzuwirken.

134.12 Gruppe History

134.12.1 Variable DEFAULT_HISTORY_LENGTH

Anzahl der Einträge, die in der History angezeigt werden.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

134.12.2 Variable HISTORY_HIDE_DATE

Wenn Sie diese Variable auf den Wert „yes“ setzen, bekommen nur Benutzer mit Zugriff auf die Superuser-Rolle in der History Datumseinträge angezeigt.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

134.12.3 Variable HISTORY_HIDE_DATE_SU

Mit dieser Variable können Sie die Datumseinträge in der History vor allen Usern verbergen. Benutzer mit Superuser-Zugriff sind eingeschlossen.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

13.4.13 Gruppe I18N

13.4.13.1 Variable DEFAULT_LOCALE

Default-Lokalisierungs-Einstellung.

Der Default-Wert ist C.

13.4.13.2 Variable LANGUAGE

Ländercodes, z.B. DE oder UK.

13.4.13.3 Variable SYSTEM-LANGUAGE

Die ausgewählte Systemsprache.

13.4.13.4 Variable SYSTEM_LOCALE

Lokalisierungs-Einstellung des Systems.

13.4.13.5 Variable USER_LOCALE

Lokalisierungs-Einstellungen der Benutzer.

13.4.14 Gruppe Imperia Develop Session

13.4.14.1 Variable ADDR_AUTH

Mit dieser Variablen definieren Sie den Modus, in dem pack_ipaddr arbeitet. Mögliche Werte sind „PROXY“ oder „FIX“ .

13.4.14.2 Variable CRYPT_ALGO

Der Verschlüsselungs-Algorithmus für die Cookies des Session Managements auf dem Produktionsserver.



Hinweis:

Für weitere Informationen siehe Crypt::CBC(3).

Der Default-Wert ist DES_PP.

13.4.15 Gruppe Datum und Zeit

13.4.15.1 Variable CALENDAR_DATE_FORMAT

Mit dieser Variablen konfigurieren Sie das Ausgabeformat mit dem Imperia Kalender-Widget generierter Datumsangaben. Bestandteile des Datums geben Sie dabei in Großbuchstaben an, Bestandteile der Uhrzeit in Kleinbuchstaben. Dazu ein Beispiel:

```
"CALENDAR_DATE_FORMAT" = "YYYY/MM/DD hh:mm"
```

Der Default ist das Format DD.MM.YYYY hh:ii.

13.4.15.2 Variable CALENDAR_TIME_FORMAT

Diese Variable dient zur Festlegung des Zeitformats von Datumsangaben, die Sie mit dem Imperia Kalender-Widget generieren. Sie können zwischen 12- und 24-stündigem Zeitformat wählen, indem Sie die Variable auf 12 bzw. 24 setzen.

13.4.15.3 Variable DATE_COMBO_MINUTES_STEP

Bei der Verwendung von Dropdownfeldern zur Eingabe von Datums- und Zeitangaben kann hier das Raster in Minuten angegeben werden, in dem der Benutzer die Zeit einstellen kann. Beispielsweise kann der Benutzer bei der Angabe "15" nur noch folgende Minutenwerte wählen: 0, 15, 30 und 45.

Ist eine Minutenangabe in einem Dokument bereits erfasst, die nicht ins Raster passt, wird auf den nächstgelegenen Rasterwert gerundet.



Wichtig:

Ist ein Wert für DATE_COMBO_MINUTES_STEP vergeben, können im Javascript-Widget keine Zeiteinstellungen (Stunden und Minuten) mehr gemacht werden.

```
"DATE_COMBO_MINUTES_STEP" = "10"
```

134.154 Variable DATE_COMBO_YEARS_LIMIT

Werden Dropdownfelder zur Datumseingabe benutzt, kann man durch Setzen dieser Konfigurationsvariable mit dem Wert „1“ die Auswahlmöglichkeiten im Jahr-Dropdown einschränken. Standardmäßig wird im Jahres-Dropdown eine Liste vom aktuellen Jahr bis 15 Jahre in die Zukunft ausgegeben. Dieser Bereich lässt sich verändern. Bei einer Angabe von „2“ werden bei einem neuen Dokument z.B. nur das aktuelle Jahr und die kommenden beiden Jahre im Dropdown angeboten.

Liegt beim Bearbeiten einer bereits bestehenden Seite das bestehende Datum plus der angegebene Bereich in der Vergangenheit, so wird ganz oben zusätzlich das aktuelle Jahr angeboten.

```
"DATE_COMBO_YEARS_LIMIT" = "2"
```

134.155 Variable DISPLAY_TIME_ZONE

Standardmäßig werden Zeit- und Datumsangaben in der Imperia-Oberfläche (also z.B. Freischalt- oder Löschdatum) passend zur Zeitzone angezeigt, die der Nutzer im Betriebssystem eingestellt hat. Ist dies unerwünscht, kann diese Umrechnung mit folgender Angabe in der Datei site/config/system.conf deaktiviert werden:

```
"DISPLAY_TIME_ZONE" = "UTC"
```

134.16 Gruppe IWE

134.16.1 Variable EWE_COMPAT_MODE

Wenn Sie diese Variable auf „1“ setzen, werden die EWE-Templateaufrufe als IWE-Aufrufe interpretiert.

134.16.2 Variable TEMPLATE_EWE

Wenn Sie diese Variable auf „yes“ setzen, ist das EWE Modul aktiviert.

Für die Verwendung des IWE sollten Sie diese Variable auf „no“ setzen oder auskommentieren.

134.17 Gruppe Link-Checker

134.17.1 Variable LINK_CHECK

Link-Checks mit HTTP-Requests aktivieren.

134.17.2 Variable NOPROXY

Ohne Proxy.

134.17.3 Variable PROXY_FTP

FTP-Proxy.

134.17.4 Variable PROXY_HTTP

HTTP-Proxy.

13.4.18 Gruppe Lock Constants

13.4.18.1 Variable LOCK_EX

Interner Wert für das File-Locking.

Der Default-Wert ist 2.

13.4.18.2 Variable LOCK_NB

Interner Wert für das File-Locking.

Der Default-Wert ist 4.

13.4.18.3 Variable LOCK_SH

Interner Wert für das File-Locking.

Der Default-Wert ist 1.

13.4.18.4 Variable LOCK_UN

Interner Wert für das File-Locking.

Der Default-Wert ist 8.

13.4.19 Gruppe MDB Counter



Wichtig:

*Mit Imperia-Version 8.5 löst das Media-Asset-Management die Mediendatenbank ab.
Diese wird jedoch aus Kompatibilitätsgründen vorübergehend weiter unterstützt.*

13.4.19.1 Variable DISABLE_MDB_REFCOUNT

Setzen Sie diese Variable auf „1“ , um den Referenzzähler der MDB abzuschalten.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

13.4.19.2 Variable LIVE_CLEANUP

Setzen Sie diese Variable auf „1“ , um die automatische Löschung unreferenzierter MDB-Objekte auf dem Zielsystem zu aktivieren.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

13.4.19.3 Variable REINDEX_MDB_FREQUENCY

Die Frequenz, in der die Indexerstellung in der Mediendatenbank ausgeführt wird. Die Angabe erfolgt in Sekunden.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

Der Default-Wert ist 24*60*60.



Hinweis:

Ab Imperia 8.6 wird diese Variable nicht mehr verwendet, da das automatische Reindizieren der Mediendatenbank deaktiviert wurde.

13.4.20 Gruppe Media-Asset-Management

13.4.20.1 DISABLE_8BIT_FILENAME

Setzen Sie die system.conf-Variable DISABLE_8BIT_FILENAME auf den Wert "1" um die Umschreibung von 8-bit-Datei- und Verzeichnisnamen in einen "sicheren" Zeichensatz beim MAM-Upload zu erzwingen.

B4202 Variable MAM_CAT_ORDER_BY_NAME

Aktivieren Sie die alphabetische Sortierung des Asset-Rubrikenbaums, indem Sie diese Variable auf 1 setzen.

B4203 Variable MAM_CATEGORY_TEMPLATE

Geben Sie systemweit den Namen Ihres unter site/templates gespeicherten Templates zur Erstellung von Asset-Rubriken an. Die Angabe erfolgt ohne Dateiendung. Fehlt diese Variable, erfolgt das Erstellen von Asset-Rubriken mit dem Standarddialog für die Rubrikerstellung.

B4204 Variable MAM_DISABLE

Setzen Sie diese Variable auf 1, wenn Sie das Media-Asset-Management deaktivieren und stattdessen die Imperia-Mediendatenbank verwenden möchten.



Wichtig:

*Mit Imperia-Version 8.5 löst das Media-Asset-Management die Mediendatenbank ab.
Diese wird jedoch aus Kompatibilitätsgründen vorübergehend weiter unterstützt.*

Das MAM ist dann weiterhin über das Menü verfügbar, und Sie können es weiterhin befüllen und administrieren, aber es erfolgt beispielsweise keinerlei Speicherung von Referenzen über das MAM. Um die Mediendatenbank wieder nutzen zu können, müssen Sie manuell einen Untermenüpunkt erstellen. Lesen Sie hierfür Abschnitt 10.4.15 **Deaktivierung des Media-Asset-Managements** auf Seite 144.

B4205 Variable MAM_ENABLE_PUBLISH

Wenn Sie Ihre Media-Assets unabhängig von Dokumenten freischalten möchten, setzen Sie diese Variable auf 1.

B4206 Variable MAM_GET_UPLOAD_FILENAME

Diese Variable aktiviert ein zusätzliches Eingabefeld in der Upload-Box in einem Asset-Template. Vorbelegt ist dieses Feld mit dem Standard-Dateinamen aus der Asset-Rubrik. Löschen Sie die Vorbelegung und lassen Sie das Feld leer, wenn Sie den ursprünglichen Dateinamen für das Asset übernehmen möchten.

"MAM_GET_UPLOAD_FILENAME" = "1"

B4207 Variable MAM_HIDDEN_FIELDS

Diese Variable beeinflusst in der Listenansicht die Anzeige der Asset-Details und der Spalten sowie die entsprechenden Links in der Iconbox-Ansicht. Notieren Sie eine kommasseparierte Liste der Spalten, deren Anzeige Sie unterbinden möchten. Das System wertet die Liste als regulären Ausdruck aus. Dazu ein Beispiel:

"MAM_HIDDEN_FIELDS" = "directory, .*filename" => m/^directory|.*filename\$/"
"MAM_HIDDEN_FIELDS" = "directory,template" => m/^directory|template\$/"

Die folgende Tabelle listet die verfügbaren Spaltennamen auf:

Deutscher Spaltenname (Listen-Ansicht)	FeldID
- (Checkbox)	__mam_select
Zoom	__mam_zoom
Icon	__mam_icon
Titel	__mam_title
Dateiname	__mam_filename
Verzeichnis	-
Dateigröße	__mam_filesize
Details	__mam_details
Referenzen	__mam_references
Löschen	__mam_delete
Erzeugt von	__mam_createdby
Erzeugungsdatum	__mam_createdat
Template	-
Abmessungen	__mam_dimensions
Benutzeraktion	__mam_action
Benutzervarianten	__mam_variants
Archiv	__mam_archive
Import	__mam_reimport

Tabelle 13.17. Verfügbare Felder im Media-Asset-Management



Hinweis:

Das Feld **Icon** lässt sich nur in der Listen-Ansicht verbergen. Für Benutzer mit Superuser-Rolle hat das Ausblenden keinen Effekt.

13.208 MAM_JPEG_QUALITY

Mit diesem Parameter können Sie einstellen, in welcher Qualität das MAM JPEG-Varianten von Bildern erstellt. Der Parameter beeinflusst die sichtbare Qualität und die Dateigröße. Grundeinstellung ist 85; der Wertebereich von 20 bis 100. Höhere Werte führen zu besserem Aussehen, erhöhen aber auch die Dateigröße. Mit kleineren Werten kann man an der Dateigröße sparen, muss aber Qualitätseinbußen in Kauf nehmen.

```
"MAM_JPEG_QUALITY" = "75"
```

13.209 Variable MAM_LINKS_STYLE

Mit der Variablen `MAM_LINKS_STYLE` legen Sie die Darstellung der Links in den Spalten der Listen-Ansicht fest. Die Syntax lautet wie folgt:

```
"MAM_LINKS_STYLE" = "[Option]"
```

Diese drei Optionen stehen zur Auswahl:

ICON

In den Spalten erscheinen lediglich kleine Icon-Grafiken.

TEXT

In den Spalten erscheinen ausschließlich Textlinks.

ICON,TEXT

In den Spalten erscheinen sowohl Icons als auch Text. Diese Einstellung ist der Default.

342010 Variable MAM_MULTIUPLOAD_OPTIONS

Mit der Variablen MAM_MULTIUPLOAD_OPTIONS richten Sie die verfügbaren Optionen für den Mehrfachupload ein. Die Angabe der Optionen erfolgt in Form einer kommaseparierten Liste von Zahlen.

Beispiel:

```
"MAM_MULTIUPLOAD_OPTIONS" = "2,3,4,5,6,7,8,9,10"
```



Hinweis:

Ab einer Anzahl von ca. 15 Assets ist mit steigenden spürbaren Performance-Einbußen zu rechnen. Das ist natürlich nur ein Richtwert. Je nach System-Gegebenheiten kann es dazu auch schon früher kommen.

342011 Variable MAM_MULTIUPLOAD_SIZE

Die Option **Datei->Mehrfachupload** erlaubt das Anlegen mehrerer Assets in einem Arbeitsgang. Mit der Variablen MAM_MULTIUPLOAD_SIZE legen Sie die Standardanzahl hierfür fest.

Beispiel:

```
"MAM_MULTIUPLOAD_SIZE" = "5"
```

342012 Variable MAM_NAV_HEADER

Mit dieser Variablen legen Sie die Überschrift des Asset-Rubrikenbaums fest.

342013 Variable MAM_NO_EXTENDER_NORMALIZE

Beim Hochladen von Mediendateien werden Endungen von erkannten Medienformaten in die in der Datei site/config/mime.types angegebene Endung umgeschrieben. So wird beispielsweise die Datei "beispiel.jpg" im Standardfall als "beispiel.jpeg" in dem MAM abgelegt. Durch Setzen der Variablen MAM_NO_EXTENDER_NORMALIZE auf den Wert "1" in der site/config/system.conf kann das Ändern der Anhänge verhindert werden.

Bitte beachten Sie auch den Abschnitt 13.4.9.17 **Variable MIME_TYPES** auf Seite 215.

Diese Variable wurde mit Imperia Version 8.5.4 eingeführt.

342014 Variable MAM_PUBLIC_SPOOL

Ist diese Variable auf "1" gesetzt, dürfen alle Benutzer über das Skript /cgi-bin/site_mam_spool_import.pl Assets in alle MAM-Rubriken importieren. Informationen zum Skript finden Sie im Abschnitt 10.3 **Import von Assets aus einem Ordner oder einer Zip-Datei** auf Seite 136.

342015 Variable MAM_READ_ONLY_FIELDS

Mit dieser Variablen beeinflussen Sie das Details-Popup der Listenansicht. Legen Sie fest, welche Felder nicht vom Benutzer editierbar sein sollen. Per Default sind dies die Felder directory, filename und template.

342016 Variable MAM_SAVED_FILTERS

Mit dieser Variablen können Sie gespeicherte Filter für alle Benutzer zur Verfügung stellen. Die Beschreibung dazu finden Sie in Abschnitt 10.4.1 **Globale Suchfilter definieren** auf Seite 140.

342017 MAM_SPOOL_CHARSET

Ist Ihr Betriebssystem-Zeichensatz nicht UTF-8, so muss dieser bei der Verwendung des MAM-Massenimports angegeben werden, um eine korrekte Übernahme von Sonderzeichen in die Imperia-Oberfläche und Logfiles zu erzielen.

134.08 Variable MAM_SPOOL_PATH

Mit dieser Variablen können Sie steuern aus welchem Verzeichnis Zip Dateien ins MAM mit dem Script /cgi-bin/mam_spool_import.pl (siehe Abschnitt 10.3 **Import von Assets aus einem Ordner oder einer Zip-Datei** auf Seite 136) hochgeladen werden können.

```
"MAM_SPOOL_PATH" = "/my_spool"
```

134.09 Variable MAM_UNIQUE

Setzen Sie diese Variable auf 1, wenn Sie möchten, dass Imperia alle Assets sowie die zugehörigen Varianten im Dateisystem in einem Unterverzeichnis der betreffenden Rubrik abspeichert. So können Sie beispielsweise verhindern, dass sich beim Verschieben von Assets zufällig identisch benannte Asset-Dateien oder Varianten überschreiben.

134.21 Gruppe Mediendatenbank



Wichtig:

Mit Imperia-Version 8.5 löst das Media-Asset-Management die Mediendatenbank ab. Diese wird jedoch aus Kompatibilitätsgründen vorübergehend weiter unterstützt.

134.21.1 Variable MD-BROWSER-VIEW

Legen Sie den Darstellungsmodus des Verzeichnisbaums der Mediendatenbank fest. Zur Auswahl stehen „standard“, „terminal“ und „explorer“.

Der Default-Wert ist *standard*.

134.21.2 Variable MD-PICS-PER-ROW

Anzahl der Thumbnails pro Reihe.

Der Default-Wert ist 3.

134.21.3 Variable MD_ENABLE_RECURSIVE

Um Änderungen in der Mediendatenbank rekursiv auszuführen, setzen Sie diese Variable auf „1“.

134.21.4 Variable MIME_ICON_DIR

Verzeichnis mit Symbolgrafiken (gif) für MIME-Types.

134.21.5 Variable MIME_TXT_DEFAULT_EXTENDER

Entsprechend der Mime-Type Datei site/config/mime.types für Dateien vom Typ „plain/text“ ist die Dateierweiterung „.asc“, die aber nicht sehr gebräuchlich ist. Breitere Verwendung findet für Text-Dateien die Endung „.txt“. Aus diesem Grund wird beim hochladen einer Datei mit der Dateiendung „.asc“ die Dateiendung auf „.txt“ umgestellt. Vermeiden kann man dies durch setzen der Variablen **MIME_TXT_DEFAULT_EXTENDER** auf den Wert „1“.

Wenn Sie die Variable **MIME_TXT_DEFAULT_EXTENDER** auf den Wert „1“ setzen, dann wird die Standard-Dateierweiterung aus der Mime-Type Datei site/config/mime.types verwendet.

```
"MIME_TXT_DEFAULT_EXTENDER" = "1"
```

Bitte beachten Sie auch den Abschnitt 13.4.9.17 **Variable MIME_TYPES** auf Seite 215.

Diese Variable wurde mit Imperia Version 8.5.4 eingeführt.

134.21.6 Variable UPLOAD_MDB

Wenn dieser Schalter aktiviert wird, werden alle dem Dokument zugeordneten MDB-Objekte publiziert.

13422 Gruppe One-Click-Edit

134221 Variable ONECLICKEDIT

Legen Sie systemweit fest, wie One-Click-Edit etwaige Kopien eines Dokuments behandeln soll. Der Defaultwert, *DOCUMENT*, stellt OCE auf dokumentbasierte Verarbeitung ein. Setzen Sie die Variable auf *FILE*, wenn Sie beispielsweise einzelne Kopien eines Dokuments löschen möchten.



Hinweis:

Wenn Sie das Verhalten der OCE-Toolbar auf Dokumentenebene variabel einstellen möchten, empfiehlt sich die Festlegung über das Template. Lesen Sie hierfür den Abschnitt One-Click-Edit in der Syntaxreferenz im Kapitel Templates des Programmierhandbuchs.

13423 Gruppe PHP

134231 Variable PHP_CLI

Typ der verwendeten PHP Version. Setzen Sie diese Variable auf einen wahren Wert, wenn Sie die CLI-Variante des PHP-Interpreters im Einsatz haben. Per Default geht das System von der CGI-Variante aus.

134232 Variable PHP_PATH

Pfad zum PHP-Interpreter.

13424 Gruppe Publisher

134241 Variable DEFAULT_VERBOSITY

Für ausführlichere Ausgaben beim Freischalten setzen Sie diese Variable auf „1“.

134242 Variable DISABLE_PUBLISHER_WARNINGS

Setzen Sie diese Variable auf „1“, um die Ausgabe von Warnmeldungen zu unterdrücken.

134243 Variable INFORM_ROLE_ON_PUBLISH

Wenn Sie möchten, dass beim Freischalten in Imperia Benachrichtigungen verschickt werden, setzen Sie diese Variable auf „1“.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

134244 Variable SERIALIZER

Die Serialisierungsmethode, die der Publisher nutzt. Mit „DUMPER“ wählen Sie das Modul Data::Dumper(3), ohne diese Variable Storable. Bei jedem anderen Wert sucht das System nach einem Plug-In dieses Namens im Verzeichnis site/modules/core/Dynamic/Serialize Ihrer Imperia-Installation.



Achtung:

Änderungen an dieser Variablen können zu Datenverlusten führen.

Der Default-Wert ist *Storable*.

134245 Variable UPLOAD_OBJECTS

Mit dieser Variablen kontrollieren Sie, welche Objekte Imperia ausliefert. Mögliche Werte sind „ALL“, „LINKS“ und „“.

13425 Gruppe Registrierungsschlüssel

134251 Variable REG_DEMO

Demo-Registrierungsschlüssel.

134252 Variable REG_KEY

Ihr Imperia Registrierungsschlüssel.

134253 Variable REG_NAME

Ihr Imperia Registrierungsname.

13426 Gruppe Reparsen

134261 Variable REPARSE_ACTION_CHECKED

Wählen Sie eins der folgenden Ziele für Dokumente, die reparst werden: „live“, „workflow“ und „pending“. Diese Einstellung kann von Benutzern geändert werden. Per Default ist die Option aktiviert.

134262 Variable REPARSE_ACTION_FIXED

Wählen Sie eins der folgenden Ziele für Dokumente, die reparst werden: „live“, „workflow“ und „pending“. Diese Einstellung kann von Benutzern nicht geändert werden.

134263 Variable REPARSE_SOURCE_CHECKED

Mit dieser Variablen können Sie bestimmen, welche Quelfelder vorausgewählt sind. Zur Auswahl stehen „live“ und „pending“ (bei Mehrfachauswahl durch Komma trennen). Mit einer leeren Zeichenkette „ legen Sie fest, dass es keine vorausgewählten Werte gibt. Diese Einstellung kann von den Benutzern geändert werden. Per Default ist die Option aktiviert.

134264 Variable REPARSE_SOURCE_FIXED

Mit dieser Variablen können Sie die Quelle auswählen. Zur Auswahl stehen „live“ und / oder „pending“ (bei Mehrfachauswahl durch Komma trennen). Mit einer leeren Zeichenkette „ legen Sie fest, dass es keine vorausgewählten Werte gibt. Diese Einstellung kann nicht von Benutzern geändert werden.

13427 Gruppe Rubriken

134271 Variable CATEGORY_TEMPLATE

Template zur Bearbeitung der Metainformationen Ihrer Rubriken.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

Der Default-Wert ist *Fallback-Template*.

134272 Variable DEFAULT_DIRECTORY

Standard-Verzeichnis für Rubriken.

Der Default-Wert ist /.

134273 Variable DEFAULT_FILENAME

Standard-Dateiname für Dokumente.

Der Default-Wert ist *index.html*.

13428 Gruppe Schreibtisch

134281 Variable DESKTOP_COLUMN_MODULES

Eine Liste der Module für Schreibtischspalten. Die einzelnen Elemente werden durch Semikola getrennt. Module müssen in einem Verzeichnis unterhalb von site/modules/core abgelegt sein.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

134282 Variable DESKTOP_CONFIG_MODULES

Eine Liste der Module zur Schreibtisch-Konfiguration. Die einzelnen Elemente werden durch Semikola getrennt. Module müssen in einem Verzeichnis unterhalb von site/modules/core abgelegt sein.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13429 Gruppe Server

134291 Variable SERVER-SCHEME

Protokoll-Schema. Möglich sind entweder http oder https.

134292 Variable SERVER-URL

Server URL.

134293 Variable SERVER_NAME

Der Fully Qualified Domain Name (FQDN) des Servers.

Diese Variable wurde mit Imperia Version 7.5 eingeführt.

13430 Gruppe SiteActive

134301 Variable SAVEPAGE-PASSWORD

Geben Sie hier ein Passwort ein, um „SAVEPAGE“ in site_active.pl über CGI nutzen zu können.

134302 Variable SCHEDULE

Doppelpunktseparierte Liste zusätzlicher Schedule-Dateien.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

13431 Gruppe Software Key

13431.1 Variable UCROP_KEY

Registrierungsschlüssel für Ucrop.

13431.2 Variable WORDMODUL_CLASSID

Classid für das Wordmodul.

13431.3 Variable WORDMODUL_KEY

Registrierungsschlüssel für das Wordmodul.

13431.4 Variable WORDMODUL_USER

Benutzername für das Wordmodul.

13432 Gruppe Standard-Zielsystem

13432.1 Variable live-url

Die absolute URL des Servers, z.B. `http://localhost.at.my.cool.server:8000/`.

13432.2 Variable liveftp1-directory

Der absolute Pfad zur Document Root auf dem Ziel-Server, z.B. `/usr/local/share/imperia/htdocs`.

13432.3 Variable liveftp1-ip

IP-Adresse und Port des Standard-Zielsystems. Diese Variable ist nur für die Übertragungsart FTP relevant.

13432.4 Variable liveftp1-name

Diese Variable enthält den Benutzernamen für den Verbindungsaufbau zum Standard-Zielsystem mittels FTP .

134325 Variable liveftp1-pass

Diese Variable speichert das Passwort für den Verbindungsaufbau zum Standard-Zielsystem mittels FTP.

134326 Variable liveftp1-site

In dieser Variablen ist der absolute Pfad zum site-Verzeichnis Ihrer Installation gespeichert, z.B. /usr/local/share/imperia/site.

134327 Variable liveftp1_passive_mode

Wenn diese Variable auf „ on “ gesetzt ist, wird der passive FTP-Übertragungsmodus genutzt.

134328 Variable transfer

In dieser Variable ist der aktive Dateiübertragungsmodus gespeichert. Der Default-Wert ist „ Filecopy “.

13433 Gruppe Template

134331 Variable ALLOW_TEMPLATE_FORM

Setzen Sie diese Variable auf „ 1 “, um Formulare in HTML-Templates zuzulassen.

134332 Variable ALLOW_TEMPLATE_FRAME

Setzen Sie diese Variable auf „ 1 “, um Frames in HTML-Templates zuzulassen.

134333 Variable ALLOW_TEMPLATE_SCRIPT

Setzen Sie diese Variable auf „ 1 “, um Scripting in HTML-Templates zuzulassen.

134334 Variable MD-DEFAULT-LINKS

Mit dieser Variablen legen Sie fest, welche Links im Mediendatenbankaufruf angezeigt werden. Mögliche Werte sind „ none “, „ upload, mdb “ oder „ mdb, upload “.

Der Default-Wert ist *mdb, upload*.

134335 Variable NO_ANON

Setzen Sie diese Variable auf „ 1 “, um die dynamische Nummerierung unbenannter Formularfelder zu deaktivieren.

134336 Variable SAFE_TEMPLATE

Setzen Sie diese Variable auf „ 1 “, damit externer Perl-Code (beispielsweise aus Flex-Modulen) in einer Safe(3)-Umggebung ausgeführt wird.

134337 Variable TEMPLATE_EWE

Setzen Sie diese Variable auf „ 1 “, um die Verarbeitung des EWE-Wordmoduls im Template-Prozessor zu aktivieren.

13434 Gruppe Templateprozessor

134341 Variable TEMPLATE_CHAIN

Beschreibt systemweit die Reihenfolge der vom Template-Prozessor verwendeten Module. Entfernen Sie nicht benötigte Module aus der Kette, um die Performance Ihres Systems zu steigern.



Achtung:

Die Entfernung zwingend notwendiger Module oder Änderungen der Standard-Reihenfolge können zu unerwarteten Ergebnissen bei der Verarbeitung der Templates führen.

Wenn Sie die in Imperia implementierten Funktionen für mehrsprachige Dokumente nutzen möchten, müssen Sie eine modifizierte Template-Chain definieren. Setzen Sie das Modul **Multilang** nach dem ersten "CodeInclude" und vor "Flex" in die Prozesskette und das Modul **MultilangFinish** an deren Ende.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.



Hinweis:

Die Metavariable `_imperia_template_chain` ist das rubrikweit gültige Äquivalent dieser Konfigurationsvariable.

13435 Gruppe User-Management-Einstellungen

13435.1 Variable DATABASE_STORAGE_PLUGIN

Datenhaltungs-Plug-In für das User Management. Entweder „FileStore“ oder „LDAP“.

Der Default-Wert ist *FileStore*.

13435.2 Variable SORT_ROLES

Setzen Sie diese Variable auf „IDs“, um die Rollen nach ihren IDs sortieren zu lassen.

13435.3 Variable TRANSITIVE-ACCESS

Mit dem Wert „off“ deaktivieren Sie den transitiven Zugriff.

13435.4 Variable UMLIGHT_PLUGIN

Benutzer-Plug-In.

Der Default-Wert ist *FilestoreLight*.

13436 Gruppe Workflow

13436.1 Variable CONFIRM_NEXT_STEP

Wenn Sie diese Variable auf „1“ setzen, muss jeder Workflow-Schritt bestätigt werden.

13436.2 Variable APPROVAL_DISABLE_JUMP

Wenn Sie diese Variable auf „1“ setzen, können Dokumente, die sich im Genehmigungs-Schritt befinden, nicht mehr in vorhergehende Workflowschritte versetzt werden.

Diese Variable wurde mit Imperia Version 8.0 eingeführt.

Index

Symbole

.meta, 154
 .mnu, 129
 .simple, 154
 <advancedHTMLTemplate>, 184
 <bonus>, 165
 <directory>, 169
 <directoryMatch>, 169
 <domain>, 164, 182
 <exclude>, 166
 <files>, 170
 <groupname>, 179
 <include>, 166
 <lang>, 182
 <map>, 183
 <meta>, 179
 <metaGroup>, 168
 <option>, 168, 182
 <output>, 183
 <pageMeta>, 172
 <pageMetaComment>, 167
 <resultHTMLPage>, 184
 <settype>, 164
 <standardHTMLTemplate>, 184
 <templateDynamicGroups>, 184
 [delete], 24
 [edit], 24
 __imperia_category, 154
 __imperia_children
 oce_clone, 38
 __imperia_clobber_by_id, 155
 __imperia_clobber_by_name, 155
 __imperia_created
 oce_clone, 38
 __imperia_imported, 153
 __imperia_meta_expand
 oce_clone, 38
 __imperia_modified
 oce_clone, 38
 __imperia_node_id
 oce_clone, 38
 oce_delete, 39
 oce_edit, 37
 oce_erase, 38
 oce_history, 39
 oce_info, 39
 oce_linkcheck, 39
 oce_reparse, 39
 oce_unpublish, 40
 __imperia_parent
 oce_clone, 38
 __imperia_quick_edit
 oce_edit, 37
 oce_linkcheck, 39
 __imperia_show_in_cat_filter, 88
 __imperia_uid
 oce_clone, 38

A

aberkennen
 Zugriffsrecht(e), 88
 abgleichen
 Systemdateien, 75
 Ablehnen bei ..., 65, 68
 abschalten
 User-Zugriff, 17
 ACL, 90
 ACL-gesteuerte Objekte, 90
 ACL-Verwaltung, 58
 ActiveX-Wordmodul
 Lizenzierung, 24
 additionalUserVars, 187
 ADIR, 198
 Admin-Skripte
 Archiv bereinigen, 207
 Auflistung aller Metafiles, 207
 Ausgabe der Metainformationen eines Dokuments, 208
 Ausgabe einer Liste aller Rollen, 207
 Ausgabe einer Liste von Flexmodulen, 207
 Benutzer löschen, 206
 Benutzereigenschaften ändern, 208
 Benutzern Rollen zuweisen, 207
 Benutzern Rollenzugriff entziehen, 207
 Benutzern Rollenzugriff gewähren, 207
 Counter ändern, 208
 Dokumente der Datei expiry.new hinzufügen, 206
 Eigenschaften einer Rolle ausgeben, 207
 Eigenschaften eines Benutzers ausgeben, 207
 Liste aller Benutzer anzeigen, 207
 Liste aller Templates anzeigen, 207
 Liste aller Workflows erstellen, 207
 Metainfo eines Dokuments ausgeben, 207
 nach Benutzern suchen, 208
 nach Dokumenten suchen, 208
 neue Rolle anlegen, 206
 neue Rubrik anlegen, 205
 neuen User anlegen, 206
 neues Dokument anlegen, 205
 Nicht mehr gültige Rollen aus ACLs löschen, 205
 Rolle löschen, 206
 Rolleneigenschaften ändern, 208
 Rollenhierarchie ändern, 206
 Rollenhierarchie anzeigen, 206
 Rollenzugriffsberechtigungen anzeigen, 207
 Rollenzugriffsberechtigungen eines Benutzers anzeigen, 207
 Rubrikeigenschaften ändern, 208
 Rubrikeneigenschaften ausgeben, 206
 Rubrikenrechte ändern, 205
 Rubrikenrechte anzeigen, 205
 Schreibtisch aufräumen, 206
 Status eines Dokumentes anzeigen, 208
 suchen nach Dateien, die nicht in longlive.log oder current.files enthalten sind, 206
 Unterrubriken anzeigen, 207
 verfallene Dokumente anzeigen, 206

-
- Zugriffsrechte für ein Metafile festlegen, 205
 - Zugriffsrechte für ein Metafile lesen, 205
 - Zugriffsrechte für ein Template festlegen, 205
 - Zugriffsrechte für ein Template lesen, 205
 - Zugriffsrechte für einen Workflow festlegen, 205
 - Zugriffsrechte für einen Workflow lesen, 205
 - `admin_accesscat.pl`, 205
 - `admin_accessmeta.pl`, 205
 - `admin_accesstemplate.pl`, 205
 - `admin_accessworkflow.pl`, 205
 - `admin_acl_update.pl`, 205
 - `admin_addcatr.pl`, 205
 - `admin_adddoc.pl`, 205
 - `admin_addrole.pl`, 206
 - `admin_adduser.pl`, 206
 - `admin_cleandesktop.pl`, 206
 - `admin_currentfile.pl`, 206
 - `admin_delrole.pl`, 206
 - `admin_deluser.pl`, 206
 - `admin_deriverole.pl`, 206
 - `admin_expire.pl`, 206
 - `admin_getcat.pl`, 206
 - `admin_getdoc.pl`, 207
 - `admin_getrole.pl`, 207
 - `admin_getuser.pl`, 207
 - `admin_grpaccess.pl`, 207
 - `admin_grpuser.pl`, 207
 - `admin_listcat.pl`, 207
 - `admin_listflex.pl`, 207
 - `admin_listmeta.pl`, 207
 - `admin_listrole.pl`, 207
 - `admin_listtemplate.pl`, 207
 - `admin_listuser.pl`, 207
 - `admin_listworkflow.pl`, 207
 - `admin_sanitisearchive.pl`, 207
 - `admin_searchdoc.pl`, 208
 - `admin_searchuser.pl`, 208
 - `admin_seetdoc.pl`, 208
 - `admin_setcat.pl`, 208
 - `admin_setcounter.pl`, 208
 - `admin_setrole.pl`, 208
 - `admin_setuser.pl`, 208
 - `admin_statdoc.pl`, 208
 - Alarm-Verzeichnis, 23
 - Allgemeine Einstellungen, 40
 - `ALLOW_TEMPLATE_FORM`, 12
 - `ALLOW_TEMPLATE_FRAME`, 12
 - `ALLOW_TEMPLATE_SCRIPT`, 12
 - Anbindung
 - Verzeichnisdienst, 95
 - ändern
 - Rubrikreihenfolge, 59
 - Systemdienste, 23
 - Änderungen an Workflows, 102
 - Änderungsverfolgung, 51
 - CSS-Formate, 51
 - anlegen
 - Freischalt-Trigger, 25
 - Metadatei, 61
 - Rolle, 89
 - Rubrik, 55
 - Systemdienst, 21
 - Unterrubrik, 56
 - User, 92
 - Workflow, 103
 - Workflow-Schritt, 106
 - Anlegen von Rubriken im Media-Asset-Management, 132
 - Anmeldekennung
 - Zielsystem, 20
 - Anmelden
 - erstmaliges, 17
 - Anmeldung
 - LDAP, 95
 - Annotate, 111
 - anzeigen
 - Menüstruktur, 131
 - SQL-Löschen-Statements, 30
 - SQL-Statements, 30
 - Anzeigen bei ..., 65, 68
 - Archiv-Bereiniger, 111
 - ASCII, 153
 - ASCII-Import, 155
 - Asset Rubrikenparameter
 - `_imperia_show_in_cat_filter`, 88
 - Asset-Aktionen, 110
 - Asset-Rubriken
 - allgemeine Informationen, 132
 - Rubriken-Eigenschaften, 85
 - Assets importieren, 136
 - Attribute
 - Imperia-Tabellen, 204
 - Aufbau
 - Imperia-Tabellen, 198
 - Import-Daten, 155
 - Aufbau von Workflow-Schritten, 102
 - Aufbau von Workflows, 102
 - Ausgabe von Meldungen bei der Indexerstellung, 178
 - Auswahl
 - Dokumentenhaltung, 28
 - Auswahlkriterium, 93
 - AUTH_PLUGIN, 17
 - Superuser, 18
 - Auto-Expiry, 111
 - Auto-Publisher, 111

B

- Basisverzeichnis, 23
- bearbeiten
 - Codeinclude, 72
 - Flexmodul, 67
 - Freischalt-Trigger, 26
 - Menü, 130
 - Metadatei, 62
 - Rolle, 91
 - Rollendaten, 91
 - Rubrik, 56
 - Slotmodul, 69
 - Template, 64
 - User, 93

-
- Userdaten, 93
 - Workflow, 104, 109
 - Bearbeiten**, 112
 - Bearbeiten und Weiterleiten, 113
 - Bearbeiten von Rubriken im Media-Asset-Management, 134
 - Bearbeitungsansicht der Konfigurationsdateien, 43
 - Bedienung**
 - Grid, 105
 - Bedingung**
 - OCE, 36
 - Benutzer-Account**
 - LDAP, 95
 - Benutzerprofil**
 - Trigger, 26
 - BerkeleyDB**, 7
 - Beschreibung**, 128
 - Rubriken-Informationen, 86
 - bestimmen**
 - Metadatei, 87
 - Rubriken-Eigenschaften, 85
 - Template, 87
 - Betriebssystem(e)**
 - Installation, 5
 - blacklist**, 167, 181
 - bonus_date_field**, 192
 - bonus_date_max**, 192
 - bonus_date_period**, 192
 - bonus_date_step**, 192
 - Button-Leiste**, 105

 - C**
 - Cache-Größe für Index bestimmen, 176
 - Cache-Verzeichnis der Indizierung, 176
 - cacheDir**, 176
 - cacheSize**, 176
 - Cat2DocMeta-Plug-In**, 113
 - CATDOC**, 167
 - CATPPT**, 168
 - CGI-BIN**, 13
 - CGI-DIR**, 13
 - chmap**, 175
 - Classpath**
 - Java-VM, 41
 - Clone-Plug-In**, 114
 - Codeinclude**
 - bearbeiten, 72
 - hochladen, 70
 - löschen, 72
 - neu erstellen, 70
 - Codeinclude-Dateien**, 69
 - Codeinclude-Management**, 69
 - öffnen, 69
 - Codeinclude-Verzeichnis**
 - erstellen, 70
 - löschen, 72
 - Codeincludes**, 69
 - compress**, 176
 - Cookie**
 - Gültigkeit, 14
 - COOKIE_TTL**, 14
 - copies**
 - oce_delete, 39
 - CSS-Formate**
 - Änderungsverfolgung, 51
 - CURRENT_PAGE**, 190

 - D**
 - Daemon**, 149
 - UNIX®, 149
 - Windows®, 149
 - Windows®-Systemdienst, 149
 - data**, 153
 - dataDir**, 177, 185
 - dataTempDir**, 177
 - Dateien beim Indizieren einschließen, 166
 - Dateien vom Indizieren ausschließen, 166
 - Dateiliste**, 60, 63, 76, 103
 - Dateiname**
 - Rubriken-Informationen, 86
 - Dateinamen als Index-Kriterium**, 170
 - Dateirechte**, 75
 - Dateisystem**, 6
 - Dateitypen zur Indizierung festlegen**, 164
 - Datenbank**
 - Einschränkungen, 27
 - Unterschieden, 27
 - Datenbank(en)**
 - Dokumentenhaltung, 27
 - Datenbank-Hostname**
 - Dokumentenhaltung, 29
 - Datenbank-Name**
 - Dokumentenhaltung, 29
 - Datenbank-Server-Port**
 - Dokumentenhaltung, 29
 - Datenbank-User**
 - Dokumentenhaltung, 29
 - DB_File**, 7
 - DBD::SQLite**, 7
 - DBI**, 31
 - DBI-Connect-String**
 - Dokumentenhaltung, 29
 - DBI-Tracing-Dateiname**
 - Dokumentenhaltung, 29
 - DBI-Tracing-Level**
 - Dokumentenhaltung, 29
 - Debugging-TTY**
 - Dokumentenhaltung, 30
 - DEFAULT_CHARSET**, 41
 - del_develop**
 - oce_delete, 39
 - DEMOMODE**, 13
 - Demomodus**, 13
 - Description**, 65, 67
 - Dienstnummer**, 22
 - directory**
 - oce_delete, 39
 - oce_unpublish, 40
 - dirGroup**, 173
 - disableUserSort**, 191
 - DocSelector**

-
- Dialogfenster-Optionen, 116
 - Konfiguration, 117
 - DocSelector-Plug-In, 116
 - Document Root, 20
 - DOCUMENT-ROOT, 13
 - documentDir, 177
 - Dokument-ID, 154
 - Dokumentenhaltung
 - Auswahl, 28
 - benutzerdefinierte Schlüssel, 29
 - Datenbank(en), 27
 - Datenbank-Hostname, 29
 - Datenbank-Name, 29
 - Datenbank-Server-Port, 29
 - Datenbank-User, 29
 - DBI-Connect-String, 29
 - DBI-Tracing-Dateiname, 29
 - DBI-Tracing-Level, 29
 - Debugging-TTY, 30
 - File65, 28
 - Konfiguration, 28
 - LongReadLen, 29
 - LongTruncOk, 29
 - MySQL, 28
 - ODBC_MSSQL, 28
 - Oracle, 28
 - Passwort, 29
 - PostgreSQL, 28
 - SID, 30
 - Statements, 30
 - Treiber-Optionen, 30
 - Typ "TEXT", 29
 - Dokumentverzeichnis für Indizierung bestimmen, 177
 - Domänen in Suchanfragen, 182
 - duplicieren
 - Workflow-Schritt, 107
 - Durchsuchen ..., 62, 64, 66, 68
 - dynamic.conf, 73
 - Dynamische Module, 73

 - E**
 - editieren
 - Codeinclude, 72
 - Editor
 - Konfigurationsdateien, 43
 - Eigenschaften
 - Workflow-Schritt, 106
 - Einbindung von Stopwortlisten in die Indexkonfiguration, 167
 - Einbindung von Stopwortlisten in die Suchkonfiguration, 181
 - eingeben
 - Rubriken-Informationen, 86
 - Einschränkungen
 - LDAP, 94
 - einstellen
 - Zugriffsrechte, 108
 - Email-Adresse, 93
 - entfernen
 - Zugriffsrecht, 109

 - F**
 - Epoche, 38, 153
 - Eraser, 118
 - Ergebnistemplate, 184
 - Erstellung eines neuen Codeinclude-Verzeichnisses, 70
 - Erstellung eines neuen Codeincludes, 70
 - erstmaliges
 - Anmelden, 17
 - erzeugen
 - Flexmodul, 66
 - Slotmodul, 68
 - Tabellen, 30, 34
 - EWE - Easy Web Editor
 - Lizenzierung, 24
 - Executable
 - Java-VM, 41
 - Exists, 118
 - Expiry, 118

-
- passiv, 20
 - FTP-Transfer, 19
 - fts.conf, 181

 - G**
 - Genehmigung, 119
 - Genehmigung mit Link-Check, 119
 - gewählte Rollen, 88
 - Gewichtung der Volltextsuche beeinflussen, 165
 - gidnumber, 98
 - Globale Rechte, 90
 - Globale Suchfilter im Media-Asset-Management, 140
 - Grid, 104
 - Bedienung, 105
 - Button-Leiste, 105
 - Plug-In-Leiste, 105
 - Verwendung, 14
 - Voraussetzungen, 14
 - groupBool, 188
 - Gruppierungs-Schema, 98
 - Gültigkeit
 - Cookie, 14

 - H**
 - Hauptmenü
 - Menüpunkt hinzufügen, 130
 - Hauptrubrik, 55
 - Hermes, 149
 - b, 151
 - d, 151
 - e, 151
 - h, 151
 - i, 151
 - k, 151
 - p, 151
 - s, 151
 - t, 151
 - u, 151
 - z, 151
 - hermes.pid, 152
 - Import, 153
 - Log-Datei, 152
 - Optionen, 150
 - Plug-In-Tabelle, 152
 - Standard-Konfiguration, 153
 - UNIX®, 149
 - Windows®, 149
 - Windows®-Systemdienst, 149
 - Hermes-Import, 153
 - __imperia_imported, 153
 - ASCII, 153
 - data, 153
 - Dokumente ersetzen, 155
 - mask, 153
 - Metadateien, 153
 - Plug-In, 153
 - XML, 153
 - Zielrubrik, 154
 - hermes.pid, 152
 - highlight, 187

 - I**
 - If-Else, 119
 - ignoreMeta, 189
 - ImageMagick, 8
 - Imperia
 - Fehlerprotokoll, 14
 - Konfiguration, 12
 - verlinkte Objekte, 13
 - Zwischenspeichern, 14
 - Imperia-Account
 - LDAP, 95
 - Imperia-OCE-Skripte, 37
 - Imperia-Tabellen
 - Attribute, 204
 - Aufbau, 198
 - Imperia-Update, 11
 - imperia.schedule, 21, 75
 - IMPERIA_ERROR_LOG, 14
 - Import
 - Hermes, 153
 - Import von MDB-Objekten ins Media-Asset-Management, 135
 - Import-Daten
 - Aufbau, 155
 - import.conf, 153
 - index, 174
 - index.conf, 163
 - Indexeinstellungen vererben, 175
 - Indizieren anhand von Metafeldern, 171
 - Indzierung aktivieren, 174
 - Informationen
 - LDAP-Zugang, 96
 - inherit, 175
 - install.log, 6
 - install.pl, 5
 - c(gibin), 5

-
- d(ocroot), 5
 - f(orce), 5
 - h(elp), 5
 - k(eep), 5
 - l(iv)e, 5
 - n(oprompt), 5
 - p, 5
 - r(m)(emove), 5
 - rf / -fr, 5
 - s(ite), 5
 - w, 5
 - Installation**
 - Betriebssystem(e), 5
 - CGI-BIN-Verzeichnis, 9
 - Document Root, 9
 - ImageMagick, 8
 - IP-Adresse, 9
 - Perl-Link, 9
 - Protokoll, 6
 - UNIX®-Derivaten, 9
 - Update, 11
 - URL, 9
 - User-ID, 9
 - Verzeichnisse, 10
 - Voraussetzungen, 6
 - Windows® 2000 / 2003 Server / XP, 10
 - Zielsystem, 11
 - Installations-Protokoll**, 6
 - Installations-Skript**, 5
 - Installationsvoraussetzungen**
 - Volltextsuche, 161
 - Intervall**
 - Systemdienst, 21
 - IP-Adresse**
 - Zielsystem, 20
 - IWE**, 78
 - Konfiguration, 78
 - AutoDetectLanguage, 83
 - BaseHref, 80
 - BodyClass, 81
 - BodyId, 81
 - CustomStyles, 81
 - DefaultLanguage, 83
 - DocType, 80
 - FillEmptyBlocks, 83
 - FontColors, 82
 - FontFormat, 82
 - FontNames, 82
 - FontSizes, 82
 - MaxUndoLevels, 80
 - Plugin.add, 83
 - StartupFocus, 81
 - ToolbarSets, 83
 - Konfigurationsdatei, 78
 - Konfigurationsoptionen, 79
 - J**
 - Java-Plug-In, 104
 - Java-VM, 40
 - Executable, 41
 - Java-Vm
 - K**
 - Classpath, 41
 - JavaRuntime-Environment
 - Hinweis, 15
 - keepphptag, 190
 - Kerberos, 95
 - Kommentar, 90
 - Kompressions-Plug-In für Suche auswählen, 176
 - Konfiguration, 40, 42
 - Dokumentenhaltung, 28
 - Imperia, 12
 - IWE, 78
 - Konfiguration der Indexerstellung, 163
 - Konfiguration der Suchanfragen, 181
 - Konfiguration des Media-Asset-Managements, 140
 - Konfigurations-Variablen
 - ABS-CGI-BIN, 212
 - ABS-DOC-ROOT, 212
 - ADDR_AUTH, 217
 - ALLOW_PERL_UPLOAD, 213
 - ALLOW_TEMPLATE_FORM, 227
 - ALLOW_TEMPLATE_FRAME, 227
 - ALLOW_TEMPLATE_SCRIPT, 227
 - APPROVAL_DISABLE_JUMP, 228
 - ARCHIVE_DISABLE_DELETE, 214
 - ARCHIVESHREDDER, 214
 - AUTH_CHAINED_PLUGINS, 211
 - AUTH_PLUGIN, 211
 - AUTH_PLUGIN_BACKUP, 211
 - BOTTOMTABLE, 209
 - CALENDAR_DATE_FORMAT, 217
 - CALENDAR_TIME_FORMAT, 217
 - CATEGORY_DELETION_PROMPT, 209
 - CATEGORY_TEMPLATE, 225
 - CGI-BIN, 213
 - CGI-DIR, 213
 - COMMWORLD_AUTH_CHECKSESSION, 211
 - CONFIRM_NEXT_STEP, 228
 - COOKIE-DOMAIN, 215
 - COOKIE-PATH, 216
 - COOKIE_NAME, 216
 - COOKIE_TTL, 216
 - COUNTER-DIR, 214
 - CRYPT_ALGO, 217
 - DATABASE_STORAGE_PLUGIN, 228
 - DATE_COMBO_MINUTES_STEP, 217
 - DATE_COMBO_YEARS_LIMIT, 218
 - DEBUG_ArchiveCleaner, 214
 - DEFAULT_DIRECTORY, 225
 - DEFAULT_FILENAME, 225
 - DEFAULT_HISTORY_LENGTH, 216
 - DEFAULT_LOCALE, 217
 - DEFAULT_VERBOSITY, 224
 - DEMOMODE, 213
 - DESKTOP_COLUMN_MODULES, 225
 - DESKTOP_CONFIG_MODULES, 225
 - DISABLE-DATE-TOOLTIP, 209
 - DISABLE-FAV-MENU, 209
 - DISABLE-FLOCK, 213

-
- DISABLE-FORTUNE-COOKIE, 213
 - DISABLE-STREAMED-OUTPUT, 209
 - DISABLE_TOOLTIP, 210
 - DISABLE_8BIT_FILERAMES, 219
 - DISABLE_MDB_REFCOUNT, 219
 - DISABLE_PUBLISHER_WARNINGS, 224
 - DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER, 209
 - DISABLE_WRITE_TRANSMISSION_CACHE, 212
 - DISGUISE_JAVA_PLUGIN, 214
 - DISPLAY_BASE_CLASS, 210
 - DISPLAY_TIME_ZONE, 218
 - DOCTREE_PLUGIN, 214
 - DOCUMENT_ROOT, 215
 - DONT_ASK_BEFOR_E_EXPLODE_ALL_NODES, 210
 - ENABLE_FLEX_ACL, 209
 - EWE_COMPAT_MODE, 218
 - FAV_MENU_LINK_ONLY, 210
 - FILE65_DOCUMENT_DIR, 212
 - FLEXINSTANCE_SKIN, 213
 - FLEXMODULE_SKIN, 213
 - FLOCK_DEBUG, 214
 - FollowSymLinks, 214
 - FROMLIVE_DELETE_DEFAULT_LIVE, 210
 - GRID_HEIGHT, 210
 - GRID_WIDTH, 210
 - Gruppe ACLs
 - ENABLE_FLEX_ACL, 209
 - Gruppe Anzeige
 - BOTTOMTABLE, 209
 - CATEGORY_DELETION_PROMPT, 209
 - DISABLE_DATE_TOOLTIP, 209
 - DISABLE_FAV_MENU, 209
 - DISABLE_STREAMED_OUTPUT, 209
 - DISABLE_TOOLTIP, 210
 - DISABLE_PUBLISHER_WARNINGS_FOR_FAILED_TRIGGER, 209
 - DISPLAY_BASE_CLASS, 210
 - DONT_ASK_BEFOR_E_EXPLODE_ALL_NODES, 210
 - FAV_MENU_LINK_ONLY, 210
 - FROMLIVE_DELETE_DEFAULT_LIVE, 210
 - GRID_HEIGHT, 210
 - GRID_WIDTH, 210
 - IBS_PAGE, 210
 - IMPERIA_SKIN_PATH, 210
 - LATCHING, 210
 - LINKTOOL_DISPLAY_ASSETS, 210
 - MAIN_MENU_LINK_ONLY, 211
 - MD_WINDOW_HEIGHT, 211
 - MD_WINDOW_WIDTH, 211
 - NO_NPH, 211
 - POPUP_WINDOW_HEIGHT, 211
 - POPUP_WINDOW_WIDTH, 211
 - TOPTABLE, 211
 - Gruppe Authentifizierung
 - AUTH_CHAINED_PLUGINS, 211
 - AUTH_PLUGIN, 211
 - AUTH_PLUGIN_BACKUP, 211
 - COMMWORLD_AUTH_CHECKSESSION, 211
 - Gruppe Datenhaltungs-Plug-In Datei
 - FILE65_DOCUMENT_DIR, 212
 - Gruppe Datum und Zeit
 - CALENDAR_DATE_FORMAT, 217
 - CALENDAR_TIME_FORMAT, 217
 - DATE_COMBO_MINUTES_STEP, 217
 - DATE_COMBO_YEARS_LIMIT, 218
 - DISPLAY_TIME_ZONE, 218
 - Gruppe Dynamische Variablen
 - ABS_CGI_BIN, 212
 - ABS_DOC_ROOT, 212
 - Gruppe Exporter
 - DISABLE_WRITE_TRANSMISSION_CACHE, 212
 - OPEN_3_USE_SYSREAD, 212
 - SSH_CLIENT, 213
 - SSH_FILED, 213
 - SSH_OPTIONS, 212
 - XALAN_CLASSPATH, 212
 - XSLT_PLUGIN, 212
 - XSLTPROC_PATH, 212
 - Gruppe Flexmodule
 - ALLOW_PERL_UPLOAD, 213
 - FLEXINSTANCE_SKIN, 213
 - FLEXMODULE_SKIN, 213
 - Gruppe Grundeinstellungen
 - ARCHIVE_DISABLE_DELETE, 214
 - ARCHIVESHREDDER, 214
 - CGI_BIN, 213
 - CGI_DIR, 213
 - COUNTER_DIR, 214
 - DEBUG_ArchiveCleaner, 214
 - DEMOMODE, 213
 - DISABLE_FLOCK, 213
 - DISABLE_FORTUNE_COOKIE, 213
 - DISGUISE_JAVA_PLUGIN, 214
 - DOCTREE_PLUGIN, 214
 - DOCUMENT_ROOT, 215
 - FLOCK_DEBUG, 214
 - FollowSymLinks, 214
 - IMPERIA_VERSION, 214
 - IMPERIA_ERROR_LOG, 214
 - MAX_REQUEST_SIZE, 214
 - MIME_TYPES, 215
 - OPERATING_SYSTEM, 215
 - PACK_STR, 215
 - SECRET_FREQUENCY, 215
 - SECURE_UMASK, 215
 - SITE_DIR, 215
 - UMASK, 215
 - USE_STRICT, 215
 - Gruppe Hermes
 - HERMES_INCOMING_DELAY, 216
 - HERMES_RECONNECT, 216
 - MAX_AUTO_PUBLISH_DOCS, 216
 - Gruppe History
 - DEFAULT_HISTORY_LENGTH, 216
 - HISTORY_HIDE_DATE, 216
 - HISTORY_HIDE_DATE_SU, 216

-
- Gruppe HTTP Cookie
 - COOKIE-DOMAIN, 215
 - COOKIE-PATH, 216
 - COOKIE_NAME, 216
 - COOKIE_TTL, 216
 - Gruppe I18N
 - DEFAULT_LOCALE, 217
 - LANGUAGE, 217
 - SYSTEM-LANGUAGE, 217
 - SYSTEM_LOCALE, 217
 - USER_LOCALE, 217
 - Gruppe Imperia Develop Session
 - ADDR_AUTH, 217
 - CRYPT_ALGO, 217
 - Gruppe IWE
 - EWE_COMPAT_MODE, 218
 - TEMPLATE_EWE, 218
 - Gruppe Link-Checker
 - LINK_CHECK, 218
 - NOPROXY, 218
 - PROXY_FTP, 218
 - PROXY_HTTP, 218
 - Gruppe Lock Constants
 - LOCK_EX, 219
 - LOCK_NB, 219
 - LOCK_SH, 219
 - LOCK_UN, 219
 - Gruppe MDB Counter
 - DISABLE_MDB_REFCOUNT, 219
 - LIVE_CLEANUP, 219
 - REINDEX_MDB_FREQUENCY, 219
 - Gruppe Media-Asset-Management
 - MAM_CAT_ORDER_BY_NAME, 220
 - MAM_CATEGORY_TEMPLATE, 220
 - MAM_DISABLE, 220
 - MAM_ENABLE_PUBLISH, 220
 - MAM_GET_UPLOAD_FILENAME, 220
 - MAM_HIDDEN_FIELDS, 220
 - MAM_JPEG_QUALITY, 221
 - MAM_LINKS_STYLE, 221
 - MAM_MULTIUPLOAD_OPTIONS, 222
 - MAM_MULTIUPLOAD_SIZE, 222
 - MAM_NAV_HEADER, 222
 - MAM_NO_EXTENDER_NORMALIZE, 222
 - MAM_PUBLIC_SPOOL, 222
 - MAM_READ_ONLY_FIELDS, 222
 - MAM_SAVED_FILTERS, 222
 - MAM_SPOOL_PATH, 223
 - MAM_UNIQUE, 223
 - MIME_TXT_DEFAULT_EXTENDER, 223
 - UPLOAD_MDB, 223
 - Gruppe Mediendatenbank
 - DISABLE_8BIT_FILERAMES, 219
 - MAM_SPOOL_CHARSET, 222
 - MD-BROWSER-VIEW, 223
 - MD-PICS-PER-ROW, 223
 - MD_ENABLE_RECURSIVE, 223
 - MIME_ICON_DIR, 223
 - Gruppe One-Click-Edit
 - ONECLICKEDIT, 224
 - Gruppe PHP
 - PHP_CLI, 224
 - PHP_PATH, 224
 - Gruppe Publisher
 - DEFAULT_VERBOSITY, 224
 - DISABLE_PUBLISHER_WARNINGS, 224
 - INFORM_ROLE_ON_PUBLISH, 224
 - SERIALIZER, 224
 - UPLOAD_OBJECTS, 224
 - Gruppe Registrierungsschlüssel
 - REG_DEMO, 224
 - REG_KEY, 225
 - REG_NAME, 225
 - Gruppe Reparsen
 - REPARSE_ACTION_CHECKED, 225
 - REPARSE_ACTION_FIXED, 225
 - REPARSE_SOURCE_CHECKED, 225
 - REPARSE_SOURCE_FIXED, 225
 - Gruppe Rubriken
 - CATEGORY_TEMPLATE, 225
 - DEFAULT_DIRECTORY, 225
 - DEFAULT_FILENAME, 225
 - Gruppe Schreibtisch
 - DESKTOP_COLUMN_MODULES, 225
 - DESKTOP_CONFIG_MODULES, 225
 - Gruppe Server
 - SERVER-SCHEME, 226
 - SERVER-URL, 226
 - SERVER_NAME, 226
 - Gruppe SiteActive
 - SAVEPAGE-PASSWORD, 226
 - SCHEDULE, 226
 - Gruppe Software Key
 - UCROP_KEY, 226
 - WORDMODUL_CLASSID, 226
 - WORDMODUL_KEY, 226
 - WORDMODUL_USER, 226
 - Gruppe Standard-Zielsystem
 - live-directory, 226
 - live-ip, 226
 - live-name, 226
 - live-pass, 227
 - live-url, 226
 - liveftp1-site, 227
 - liveftp1_passive_mode, 227
 - transfer, 227
 - Gruppe Template
 - ALLOW_TEMPLATE_FORM, 227
 - ALLOW_TEMPLATE_FRAME, 227
 - ALLOW_TEMPLATE_SCRIPT, 227
 - MD-DEFAULT-LINKS, 227
 - NO_ANON, 227
 - SAFE_TEMPLATE, 227
 - TEMPLATE_EWE, 227
 - Gruppe Templateprozessor
 - TEMPLATE_CHAIN, 227
 - Gruppe User-Management-Einstellungen
 - DATABASE_STORAGE_PLUGIN, 228
 - SORT_ROLES, 228
 - TRANSITIVE-ACCESS, 228
 - UMLIGHT_PLUGIN, 228
 - Gruppe Workflow

APPROVAL_DISABLE_JUMP, 228
 CONFIRM_NEXT_STEP, 228
 HERMES_INCOMING_DELAY, 216
 HERMES_RECONNECT, 216
 HISTORY_HIDE_DATE, 216
 HISTORY_HIDE_DATE_SU, 216
 IBS_PAGE, 210
 IMPERIA-VERSION, 214
 IMPERIA_ERROR_LOG, 214
 IMPERIA_SKIN_PATH, 210
 INFORM_ROLE_ON_PUBLISH, 224
 LANGUAGE, 217
 LATCHING, 210
 LINK_CHECK, 218
 LINKTOOL_DISPLAY_ASSETS, 210
 live-url, 226
 LIVE_CLEANUP, 219
 liveftp1-directory, 226
 liveftp1-ip, 226
 liveftp1-name, 226
 liveftp1-pass, 227
 liveftp1-site, 227
 liveftp1_passive_mode, 227
 LOCK_EX, 219
 LOCK_NB, 219
 LOCK_SH, 219
 LOCK_UN, 219
 MAIN_MENU_LINK_ONLY, 211
 MAM_CAT_ORDER_BY_NAME, 220
 MAM_CATEGORY_TEMPLATE, 220
 MAM_DISABLE, 220
 MAM_ENABLE_PUBLISH, 220
 MAM_GET_UPLOAD_FILENAME, 220
 MAM_HIDDEN_FIELDS, 220
 MAM_JPEG_QUALITY, 221
 MAM_LINKS_STYLE, 221
 MAM_MULTIUPLOAD_OPTIONS, 222
 MAM_MULTIUPLOAD_SIZE, 222
 MAM_NAV_HEADER, 222
 MAM_NO_EXTENDER_NORMALIZE, 222
 MAM_PUBLIC_SPOOL, 222
 MAM_READ_ONLY_FIELDS, 222
 MAM_SAVED_FILTERS, 222
 MAM_SPOOL_CHARSET, 222
 MAM_SPOOL_PATH, 223
 MAM_UNIQUE, 223
 MAX_AUTO_PUBLISH_DOCS, 216
 MAX_REQUEST_SIZE, 214
 MD-BROWSER-VIEW, 223
 MD-DEFAULT-LINKS, 227
 MD-PICS-PER-ROW, 223
 MD-WINDOW-HEIGHT, 211
 MD-WINDOW-WIDTH, 211
 MD_ENABLE_RECURSIVE, 223
 MIME_ICON_DIR, 223
 MIME_TXT_DEFAULT_EXTENDER, 223
 MIME_TYPES, 215
 NO-NPH, 211
 NO_ANON, 227
 NOPROXY, 218
 ONECLICKEDIT, 224
 OPEN_3_USE_SYSREAD, 212
 OPERATING-SYSTEM, 215
 PACK_STR, 215
 PHP_CLI, 224
 PHP_PATH, 224
 POPUP_WINDOW_HEIGHT, 211
 POPUP_WINDOW_WIDTH, 211
 PROXY_FTP, 218
 PROXY_HTTP, 218
 REG_DEMO, 224
 REG_KEY, 225
 REG_NAME, 225
 REINDEX_MDB_FREQUENCY, 219
 REPARSE_ACTION_CHECKED, 225
 REPARSE_ACTION_FIXED, 225
 REPARSE_SOURCE_CHECKED, 225
 REPARSE_SOURCE_FIXED, 225
 SAFE_TEMPLATE, 227
 SAVEPAGE-PASSWORD, 226
 SCHEDULE, 226
 SECRET_FREQUENCY, 215
 SECURE_UMASK, 215
 SERIALIZER, 224
 SERVER-SCHEME, 226
 SERVER-URL, 226
 SERVER_NAME, 226
 SITE-DIR, 215
 SORT_ROLES, 228
 SSH-CLIENT, 213
 SSH-FILED, 213
 SSH-OPTIONS, 212
 SYSTEM-LANGUAGE, 217
 SYSTEM_LOCALE, 217
 TEMPLATE_CHAIN, 227
 TEMPLATE_EWE, 218, 227
 TOPTABLE, 211
 transfer, 227
 TRANSITIVE-ACCESS, 228
 UCROP_KEY, 226
 UMASK, 215
 UMLIGHT_PLUGIN, 228
 UPLOAD_MDB, 223
 UPLOAD_OBJECTS, 224
 USE_STRICT, 215
 USER_LOCALE, 217
 WORDMODUL_CLASSID, 226
 WORDMODUL_KEY, 226
 WORDMODUL_USER, 226
 XALAN_CLASSPATH, 212
 XSLT_PLUGIN, 212
 XSLTPROC_PATH, 212
 Konfigurationsdatei
 Suche, 181
 Suchindex, 163
 Konfigurationsdateien, 42
 anzeigen lassen, 42
 bearbeiten, 43
 Bearbeitungsansicht, 43
 editierbar machen, 44
 Editor, 43
 Übersicht, 42

-
- Kontext beim Suchergebnis anzeigen lassen, 186
 Konventionen, 1
 Kopien von Dokumenten erzeugen, 114
 kopieren
 Systemdateien, 75
- L**
- LATCHING, 14
 LDAP, 94-95, 197
 ADIR, 197
 Anmeldung, 95
 Attribute, 96
 Benutzer-Account, 95
 Benutzeranbindung, 99
 Einschränkungen, 94
 Imperia als Frontend, 100
 Imperia-Account, 95
 Konfiguration, 95
 PAM, 197
 Passwort-Kodierung, 97
 Passwort-Verschlüsselung, 97
 Rollen-Attribute, 98
 Rollen-Container, 98
 Schreibzugriff, 99
 Superuser-Rolle, 99
 Superuser-Zugang, 99
 User-Container, 96
 LDAP-Filter, 98
 Rollen-Identifikation, 98
 LDAP-Perl-Modul, 94
 LDAP-Zugang
 Informationen, 96
 ldap.conf, 95-96
 Link, 128-129
 Link-Checker, 41
 Link-Text, 128
 Links, 61, 63
 Linktext, 129
 Linux
 MySQL, 34
 Oracle, 33
 PostgreSQL, 32
 Liste
 RDBMS-Treiber, 32
 Liste einschränken, 91, 93
 Lizenzierung
 ActiveX-Wordmodul, 24
 erworbenen Funktionen, 19
 EWE - Easy Web Editor, 24
 Konfiguration, 24
 Module, 24
 uCrop, 24
 Lizenznehmer, 19
 Log-Datei
 Hermes, 152
 Login, 93
 Login-Maske, 17
 LongReadLen
 Dokumentenhaltung, 29
 LongTruncOk
 Dokumentenhaltung, 29
- löschen
 Codeinclude, 72
 Codeinclude-Verzeichnis, 72
 Flexmodul, 67
 Freischalt-Trigger, 27
 Metadatei, 62
 Rolle, 92
 Rubrik, 60
 Slotmodul, 69
 Systemdienst, 24
 Tabellen, 30, 35
 Template, 64
 Untermenü, 131
 User, 94
 Verbindung, 107
 Workflow, 104, 109
 Workflow-Schritt, 107
 Löschen von Rubriken im Media-Asset-Management, 134
 lösen
 Verbindung, 107
- M**
- Maintenance Mode, 17
 MaintenanceNoLogin, 17
 MAM
 Flash-Mehrfachupload, 147
 mam_spool_import.pl, 137
 mask, 153
 match, 180
 MAX_PAGE_LIST, 190
 MAX_PAGES, 190
 maxDepth, 178
 Media-Asset-Management
 Allgemeines, 132
 Allgemeines zu Asset-Rubriken, 132
 Asset-Rubrik anlegen, 132
 Asset-Rubrik bearbeiten, 134
 Asset-Rubrik löschen, 134
 globale Suchfilter, 140
 Import von Objekten aus der Mediendatenbank, 135
 Konfiguration, 140
 Template, 147
 Zip-Import, 136
 Mediendatenbank-Objekte ins Media-Asset-Management importieren, 135
 Mehr-Augen-Prinzip, 119
 memberuid, 98
 memuseControl, 178
 Menü
 bearbeiten, 130
 Zugriff, 130
 Zugriffsrechte, 130
 Menü bearbeiten, 130
 Menüabkürzungen, 128
 Menüpunkt hinzufügen
 Hauptmenü, 130
 Untermenü, 130
 Menüstruktur, 131
 Meta-Import, 156

-
- Meta-Information Bearbeiten
 - Rubriken-Informationen, 86
 - Meta-Tags indizieren, 172
 - Metadatei
 - anlegen, 61
 - bearbeiten, 62
 - bestimmen, 87
 - hochladen, 61
 - löschen, 62
 - Metadateien, 60, 153
 - Rechte, 60
 - Metadateien und Templates auswählen
 - Rubriken-Eigenschaften, 85
 - Metaedit, 120
 - Metafeld
 - Trigger, 26
 - Metafelder als Ausschlusskriterium, 171
 - Metafelder indizieren, 179
 - Metafelder nicht indizieren, 171
 - Metagruppen dynamisch benennen, 180
 - metaKey, 171
 - Metakommentare indizieren, 167
 - Metasetter, 121
 - MetaSplice, 122
 - minValidCharacters, 187
 - Module
 - Lizenzierung, 24
 - MS-SQL, 27
 - Mülleimer, 123
 - Multilanguage, 123
 - multivalued, 98
 - MySQL, 27-28
 - Linux, 34
 - Windows, 33
- N**
- Name, 65, 67, 93
 - Rubriken-Informationen, 86
 - Namespace
 - OCE, 35
 - neu
 - Zielsystem, 25
 - Neue Hauptrubrik, 55
 - NodeID, 154
 - Notification
 - Zielformat, 26
 - Notification File, 23
 - Null, 123
- O**
- Objektklassen
 - Imperia-User, 100
 - Rolle, 100
 - Obligatorische Angaben, 90, 92
 - OCE, 35
 - Bedingung, 36
 - Hide, 36
 - Hilfe, 36
 - Icon, 36
 - Label, 36
 - sichtbar, 36
 - Skript hinzufügen, 37
 - Tooltip, 37
 - URL, 36-37
 - Zugriffsrecht(e), 36
 - OCE-Skript
 - hinzufügen, 37
 - OCE-Skripte
 - Imperia, 37
 - oce_clone, 38
 - __imperia_children, 38
 - __imperia_created, 38
 - __imperia_meta_expand, 38
 - __imperia_modified, 38
 - __imperia_node_id, 38
 - __imperia_parent, 38
 - __imperia_uid, 38
 - oce_delete, 38
 - __imperia_node_id, 39
 - copies, 39
 - del_develop, 39
 - directory, 39
 - filename, 39
 - oce_edit, 37
 - __imperia_node_id, 37
 - __imperia_quick_edit, 37
 - finished, 37
 - oce_erase, 38
 - __imperia_node_id, 38
 - force, 38
 - oce_history, 39
 - __imperia_node_id, 39
 - oce_info, 39
 - __imperia_node_id, 39
 - oce_linkcheck, 39
 - __imperia_node_id, 39
 - __imperia_quick_edit, 39
 - finished, 39
 - oce_reparse, 39
 - __imperia_node_id, 39
 - finished, 40
 - publish, 40
 - oce_unpublish, 40
 - __imperia_node_id, 40
 - directory, 40
 - filename, 40
 - ODBC_MSSQL, 28, 30
 - One-Click-Edit, 35
 - One-Click-Edit-Konfiguration, 35
 - OPERATING-SYSTEM, 13
 - Operation, 59
 - Optionale Angaben, 90, 92
 - Optionen
 - Hermes, 150
 - Optionen bei der Indexerstellung, 168
 - Optionen in Suchanfragen, 182
 - Options-Direktiven für Suchanfragen, 185
 - Oracle, 27-28, 30
 - Linux, 33
 - Ordner
 - Untermenü hinzufügen, 129

P

PAM, 98, 197
 Parameter, 59
 parseroptions-Direktiven für Suchanfragen, 189
 passiv
 FTP, 20
 passives FTP
 Zielsystem, 20
 Passwort
 Dokumentenhaltung, 29
 gültig, 92
 wiederholen, 92
 Zielsystem, 20
 Passwort-Kodierung, 97
 Passwort-Verschlüsselung, 97
 PDFINFO, 167
 PDFTOTEXT, 167
 PER_PAGE, 191
 percentPrecision, 186
 perl, 65, 67
 Perl-Code
 sichere Umgebung, 14
 Perl-Link, 9
 Perl-Modul
 LDAP, 94
 Perl-Modul(e)
 BerkeleyDB, 7
 DB_File, 7
 DBD::SQLite, 7
 Storable, 6
 Perl-Version, 6
 Pfad zum Excel-Parser der Suche, 167
 Pfad zum PDF-Parser der Suche 1, 167
 Pfad zum PDF-Parser der Suche 2, 167
 Pfad zum Powerpoint-Parser der Suche, 168
 Pfad zum Word-Parser der Suche, 167
 Pfade von Suchergebnissen gemappt ausgeben, 183
 phpEnable, 186
 phpPath, 186
 Plug-In, 153
 interaktiv, 109
 nicht interaktiv, 109-110
 Plug-In-Leiste, 105
 Plug-In-Tabelle, 152
 PostgreSQL, 27-28, 30
 Linux, 32
 SuSE 7.1, 32
 Windows, 33
 process_ssi, 190
 Protokoll
 Installation, 6
 Protokolle, 19
 publish
 oce_reparse, 40
 Publish, 124

Q

queryMode, 187

R

RDBMS
 spezifische Treiber, 31
 Zielformat, 26
 RDBMS-Treiber
 Aufstellung, 32
 RDN-Attribut
 Rolle, 100
 User-Eintrag, 100
 Rechte, 61, 63, 76, 103
 Datei, 75
 Metadateien, 60
 Rubrik, 58
 Workflow, 103
 Rechte entziehen (-), 59, 76
 Rechte hinzufügen (+, 76
 Rechte hinzufügen (+), 59
 Rechte setzen (=), 59, 77
 Rechtevergabe, 75
 REG_KEY, 14
 REG_KEY2, 14
 REG_NAME, 14
 Registrierungsschlüssel, 19
 regkey.bin, 19
 rekursive Indizierung aktivieren, 178
 reparsen
 Template, 73
 Reparser, 73
 resultFilterPlugin, 188
 Rolle, 128
 anlegen, 89
 bearbeiten, 91
 löschen, 92
 Rollen-ID, 92
 Rollen, 59, 89, 91, 129
 Rollen-ID, 92
 Rollendaten
 bearbeiten, 91
 Rolleneinstellungen, 91
 Rollenzugehörigkeit, 90, 93
 Rubrik
 anlegen, 55
 bearbeiten, 56
 löschen, 60
 Rechte, 58
 Reihenfolge ändern, 59
 Zugriff, 87
 Rubrik-ID, 154
 Rubrik-Metainformationen in Dokumente übertragen, 113
 Rubriken, 55, 85
 Zugriffrechte, 85
 Rubriken-ACL-Verwaltung, 58
 Rubriken-Eigenschaften
 Asset-Rubriken, 85
 bestimmen, 85
 Metadateien und Templates auswählen, 85
 Rubriken-Information, 85
 Workflow-Auswahl, 85
 Zugriffsrechte, 85
 Rubriken-Information

-
- Rubriken-Eigenschaften, 85
 - Rubriken-Informationen
 - Beschreibung, 86
 - Dateiname, 86
 - eingeben, 86
 - Meta-Information Bearbeiten, 86
 - Name, 86
 - Verzeichnis, 86
 - Rubriken-Metainformationen, 56
 - Rubriken-Metainformationen bearbeiten
 - bearbeiten, 57

 - S**
 - SAFE_TEMPLATE, 14
 - Secure Copy, 20
 - Server
 - Freischalt-Trigger, 25
 - SERVER-NAME, 13
 - SERVER-PORT, 14
 - SERVER-SCHEME, 13
 - SFTP-Transfer, 20
 - showContext, 186
 - Shredder, 124
 - SID
 - Dokumentenhaltung, 30
 - Site-Verzeichnis, 20
 - site/flex, 66
 - site/Slot, 68
 - site_hermes, 149
 - site_hermes.pl, 149
 - SiteActive-Basisverzeichnis, 23
 - SiteActive-Erstellung, 23
 - Skript(e)
 - im Template, 12
 - Slotmodul, 67
 - bearbeiten, 69
 - erzeugen, 68
 - hochladen, 68
 - löschen, 69
 - Slotmodul-Verwaltung, 68
 - Smart Database Update, 15
 - Upgrade und Wartung, 15
 - SmartMeta, 124
 - sortBy, 191
 - Speichergröße für Indexerstellung, 178
 - Speicherort des Index für Suchanfragen angeben, 185
 - Speicherort des Suchindex bestimmen, 177
 - spezifische Treiber
 - RDBMS, 31
 - Sprache
 - Standard, 92
 - SQL-Lösche-Statements
 - anzeigen, 30
 - SQL-Server
 - Windows, 34
 - SQL-Statements
 - anzeigen, 30
 - SSH, 20
 - Standard-Konfiguration
 - Hermes, 153

 - Standardfarben ändern
 - Stylesheet, 51
 - Statements
 - Dokumentenhaltung, 30
 - Steuerung der Ausgabe von Suchergebnissen, 183
 - Storable, 6
 - Struktur, 55
 - Flexmodule, 65
 - Metadateien, 60
 - Slotmodule, 67
 - Templates, 62
 - Stylesheet
 - Standardfarben ändern, 51
 - Such-Metagruppen definieren, 168
 - Suchdomänen einrichten, 164
 - Suchergebnis-Seite angeben, 184
 - Suchindex, 162
 - Suchindex erstellen, 193
 - Suchmuster für Metagruppen definieren, 180
 - Suchskripte erstellen, 194
 - Suchtemplates, 183
 - Suchtemplates mit PHP, 186
 - Suchtiefe beim Indizieren, 178
 - Symbol, 128-129
 - Symbolische Links bei der Indexerstellung, 178
 - Symbolliste, 129
 - Syntax der Volltextsuche, 162
 - System
 - Konfiguration, 42
 - Konfigurationsdateien, 42
 - System-Sprache, 41
 - system.conf, 12, 95
 - system_files.conf, 75
 - Systemdateien
 - abgleichen, 75
 - kopieren, 75
 - übertragen, 75
 - Systemdienst
 - anlegen, 21
 - Dateien ausschließen, 23
 - Intervall, 21
 - mehrere Alarm-Verzeichnisse, 23
 - mehrere Verzeichnisse überwachen, 23
 - Verzeichnisse ausschließen, 23
 - Systemdienste
 - ändern, 23
 - Anzeige sortieren, 23
 - Ausführungszeitpunkt, 22
 - Beschreibung, 22
 - Dienstnummer, 22
 - Intervall, 22
 - löschen, 24
 - SiteActive-Basisverzeichnis, 23
 - SiteActive-Erstellung, 23
 - SiteActive-Template, 23
 - SiteActive-Zielseite, 23
 - Speicherdatei, 21

 - T**
 - Tabellen

-
- erzeugen, 30, 34
 - löschen, 30, 35
 - Tamino
 - Zielformat, 26
 - Template
 - bearbeiten, 64
 - bestimmen, 87
 - Formular(e), 12
 - Frame(s), 12
 - hochladen, 64
 - löschen, 64
 - Mediendatenbank, 147
 - reparsen, 73
 - Skript(e), 12
 - Template-Prozessor, 41
 - Template-Reparser, 73
 - Template-Vorschau, 64
 - Templates, 62
 - temporäres Arbeitsverzeichnis für Suchindex bestimmen, 177
 - Time to live, 14
 - Tooltip
 - OCE, 37
 - Treiber-Optionen
 - Dokumentenhaltung, 30
 - Trigger
 - Benutzerprofil, 26
 - Freischalt-Trigger, 25
 - Immer, 26
 - Metafeld, 26
 - Zielsystem, 26
 - Trockenlauf, 30
 - Typ, 65, 67
 - Typ "TEXT"
 - Dokumentenhaltung, 29
- U**
- Übersicht
 - Konfigurationsdateien, 42
 - übertragen
 - Systemdateien, 75
 - Übertragungsprotokoll
 - Zielsystem, 26
 - Übertragungsprotokolle, 19
 - uCrop
 - Lizenzerstellung, 24
 - uid, 98
 - uidnumber, 98
 - Unicode, 41
 - Unter-Workflow, 124
 - Untermenü
 - anlegen, 128
 - löschen, 131
 - Menüpunkt hinzufügen, 130
 - Untermenü hinzufügen
 - Hauptmenü, 129
 - Unterrubrik
 - anlegen, 56
 - Update, 11
 - installieren, 11
 - upload
- Codeinclude, 70
 - Upload, 125
 - UPLOAD_OBJECTS, 13
 - URI
 - Zielsystem, 20
 - URI-Konflikte auflösen, 116
 - URL
 - OCE, 36
 - User, 89, 93
 - anlegen, 92
 - bearbeiten, 93
 - löschen, 94
 - User-Container, 96
 - User-Zugriff
 - abschalten, 17
 - Userdaten
 - bearbeiten, 93
 - Usereinstellungen, 92-93
- V**
- verbinden
 - Workflow-Schritt, 107
 - Verbindung
 - löschen, 107
 - lösen, 107
 - verbosityLevel, 178
 - verfeinertes Template für Suchanfragen angeben, 184
 - verfügbare Rollen, 87
 - verschieben
 - Workflow-Schritt, 107
 - Verwaltung
 - Workflow, 103
 - Verwendung
 - Grid, 14
 - Verzeichnis
 - Rubriken-Informationen, 86
 - Verzeichnisdienst
 - Anbindung, 95
 - Verzeichnisdienst(e), 94
 - Verzeichnisgruppen im Suchindex, 173
 - Verzeichnisnamen als Index-Kriterium, 169
 - Verzeichnisse indizieren, 169
 - Volltextsuche
 - Alternative Schreibweisen / Worttrennungen, 158
 - Anzahl der Treffer pro Seite, 191
 - Anzeige von Suchergebnissen, 159
 - Ausgabe von Meldungen bei der Indexerstellung, 178
 - Ausgabe von Suchergebnissen steuern, 183
 - automatische Selectbox für Gruppenauswahl, 184
 - Beispiel für Metagruppen, 180
 - benötigte Perlmodule, 161
 - benutzerdefinierte Sortieroptionen unterdrücken, 191
 - Berechnung der Treffer-Relevanz und Boni, 160, 165
 - Cache-Größe für Index bestimmen, 176
 - Dateinamen als Index-Kriterium, 170

-
- datumsbezogener Relevanzbonus, 192
 Default-Sortermethode der Trefferliste, 191
 Direktiven für Suchoptionen, 185, 189
 Dokumentverzeichnis bestimmen, 177
 Domänen, 162
 Domänen in Suchanfragen, 182
 dynamisch benannte Meta-Gruppen, 180
 Einbindung von Stopwortlisten in die Indexkonfiguration, 167
 Einbindung von Stopwortlisten in die Suchkonfiguration, 181
 Einführung, 157
 Ergebnis-Template angeben, 184
 Ergebnisfilter, 160, 188
 Erstellung eines Index, 193
 Gewichtung beeinflussen, 165
 Groß -und Kleinschreibung, 157
 Hervorhebung von Treffern einstellen, 187
 Highlighting in Suchergebnissen, 160
 Höchstzahl generierter Trefferseiten, 190
 Höchstzahl von Links auf Trefferseiten, 190
 Index-Speicherort für Suchanfragen angeben, 185
 Indexerstellung, 163
 indizierte Datentypen bestimmen, 164
 Indizierung aktivieren, 174
 Indizierungs-Konfiguration vererben, 175
 Kategorisierung von Suchanfragen, 158
 Kompressions-Plug-In auswählen, 176
 Konfiguration, 162
 Kontextanzeige bei Suchergebnissen, 160
 logische Operatoren, 159
 logische Verknüpfung von Metagruppen, 188
 maximale Suchtiefe, 178
 Meta-Angaben von PDF-Dateien indizieren, 172
 Meta-Gruppen, 179
 Meta-Gruppennamen festlegen, 179
 Meta-Kommentare, 167
 Meta-Tags indizieren, 172
 Metadaten beim Indexieren ignorieren, 189
 Metafelder als Ausschlusskriterium, 166, 171
 Metafelder als Indexkriterium, 166
 Metafelder als Kriterium zum Indizieren, 171
 Metafelder indizieren, 179
 Metafelder nicht indizieren, 171
 Metagruppen, 168
 Microsoft Office Dokumente indizieren, 161
 Minimalanzahl feststehender Buchstaben, 187
 nach Metainhalten filtern, 159
 Nachkommastellen für PAGE_PERCENT, 186
 Option-Blöcke, 168
 Optionen in Suchanfragen, 182
 PDF-Dokumente indizieren
 kripte für die Konfiguration, 161
 Pfad zum Excel-Parser, 167
 Pfad zum PDF-Parser 1, 167
 Pfad zum PDF-Parser 2, 167
 Pfad zum PHP-Interpreter angeben, 186
 Pfad zum Powerpoint-Parser, 168
 Pfad zum Word-Parser, 167
 Pfade in der Ausgabe mappen, 183
 phonetische Ähnlichkeit, 158
 PHP im Suchtemplate parsen, 186
 Phrasensuche, 158
 Reguläre Ausdrücke, 163
 rekursive Indizierung aktivieren, 178
 Schnellstart, 161
 Skripte für die Konfiguration, 160
 Sonderzeichen und Interpunktions, 157
 Speicher kontrollieren, 178
 Speicherort des Index bestimmen, 177
 Speicherort für Datei-Cache bestimmen, 176
 Sprachspezifische Synonyme definieren, 182
 Standard Such-Template bestimmen, 184
 Startseite der Ergebnisliste bestimmen, 190
 Steuerung der Kontextanzeige, 186
 Stopwörter, 158
 Suchbegriffe gruppieren, 159
 Suchdomains konfigurieren, 164
 Suche nach Metagruppen ohne Suchwort, 187
 Suchkonfiguration, 181
 Suchmuster für Metagruppen definieren, 180
 Suchskripte erstellen, 194
 Symbolische Links, 178
 Syntax, 162
 Syntax für Suchbegriffe, 157
 Templates, 183
 temporäres Arbeitsverzeichnis, 177
 Trefferliste sortieren, 191
 Umlaute, 158
 URL-Suche, 158
 verfeinertes Such-Template bestimmen, 184
 Verwendung mehrerer Wildcards ermöglichen, 187
 Verzeichnisnamen als Index-Kriterium, 169
 Verzeichnisse gruppieren, 173
 Verzeichnisse indizieren, 169
 Voraussetzungen, 161
 Wildcards, 157
 Wortstammsuche, 159
 Zeichensatz für Verzeichnisse festlegen, 175
 zusätzliche Variablen, 187
 Voraussetzungen
 Grid, 14
 Installation, 6
 Vorname, 93
 Vorschau
 Template, 64
- W**
- Wartung, 17
 Wartungsarbeiten, 17
 Wartungsmodus, 17
 mit Superuser-Zugriff, 18
 WebDAV, 20
 Webserver, 8
 Windows
 MySQL, 33
 PostgreSQL, 33
 SQL-Server, 34
 Windows®-Systemdienst

-
- Hermes, 149
 - Workflow**
 - anlegen, 103
 - bearbeiten, 104, 109
 - festlegen, 86
 - löschen, 104, 109
 - Rechte, 103
 - Zugriff, 103
 - Workflow-Auswahl**
 - Rubriken-Eigenschaften, 85
 - Workflow-Einstiegspunkt**, 86
 - Workflow-Plug-In**
 - Annotate, 111
 - Archiv-Bereiniger, 111
 - Asset-Aktionen, 110
 - Auto-Expiry, 111
 - Auto-Publisher, 111
 - Bearbeiten, 112
 - Bearbeiten und Weiterleitem, 113
 - Cat2DocMeta, 113
 - Clone, 114
 - Definition, 106
 - DocSelector, 116
 - Eraser, 118
 - Exists, 118
 - Expiry, 118
 - Genehmigung, 119
 - Genehmigung mit Link-Check, 119
 - If-Else, 119
 - Metaedit, 120
 - Metasetter, 121
 - MetaSplice, 122
 - Mülleimer, 123
 - Multilanguage, 123
 - Null, 123
 - Publish, 124
 - Shredder, 124
 - SmartMeta, 124
 - Unter-Workflow, 124
 - Upload, 125
 - Zufall, 127
 - Workflow-Plug-Ins**, 109
 - Workflow-Schritt**
 - anlegen, 106
 - Definition, 106
 - duplicieren, 107
 - Eigenschaften, 106
 - löschen, 107
 - verbinden, 107
 - verschieben, 107
 - Zugriffsrechte, 108
 - Workflow-Schritte**
 - Aufbau, 102
 - Workflow-Schritte)**
 - Änderungen, 102
 - Workflows**, 102
 - Aufbau, 102
 - verwalten, 103
 - WYSIWYG-Editor**, 78
- X**
- XLS2CSV, 167
 - XML**, 153
 - Zielformat, 26
 - XML-Import, 155
- Z**
- Zeichensatz für Verzeichnisse festlegen (Suche), 175
 - Zieldatei**
 - Freischalt-Trigger, 25
 - Zielsystem, 26
 - Zielformat**, 25
 - Freischalt-Trigger, 25
 - HTML, 26
 - Notification, 26
 - RDBMS, 26
 - Tamino, 26
 - XML, 26
 - Zielrubrik**
 - Hermes-Import, 154
 - Zielsystem**
 - Anmeldekennung, 20
 - Installation, 11
 - IP-Adresse, 20
 - neu, 25
 - passives FTP, 20
 - Passwort, 20
 - Trigger, 26
 - Übertragungsprotokoll, 26
 - URI, 20
 - Zieldatei, 26
 - zusätzlich, 25
 - Zip-Datei Import, 136
 - Zufall, 127
 - Zugriff**
 - Menü, 130
 - Rubrik, 87
 - Workflow, 103
 - Zugriffsrecht**
 - Bearbeiten, 108
 - entfernen, 109
 - Lesen, 108
 - Löschen, 108
 - Zugriffsrecht(e)**
 - aberkennen, 88
 - OCE, 36
 - zuweisen, 87
 - Zugriffsrechte, 90, 93
 - einstellen, 108
 - Menü, 130
 - Rubriken, 85
 - Rubriken-Eigenschaften, 85
 - Workflow-Schritt, 108
 - zusätzlich**
 - Zielsystem, 25
 - Zusätzliche Links**, 128
 - zuweisen**
 - Zugriffsrecht(e), 87
 - ZwischenSpeichern**
 - Imperia, 14