



SEBLOD®
by Octopos

Axel Tüting

2., aktualisierte und
erweiterte Auflage

SEBLOD® Handbuch

Mehr als nur das Standard-Joomla!
Das Content Construction Kit SEBLOD® in der Praxis.

- Installation, Konfiguration und Erweiterung mit bestehenden Apps
- Alles für die eigene App: Felder, Formulare, Inhaltstypen, Listen und Suchtypen
- Profiwissen: Beziehungslisten, Module, Multisites, Plug-ins und Templates

Axel Tüting
SEBLOD® Handbuch

Axel Tüting ist Fachinformatiker für Anwendungsentwicklung. 2008 gründete er die Firma time4mambo für Programmierung, Schulung und Webentwicklung mit Fokus auf Joomla! und seiner Vorgängerversion Mambo. Der profilierte Joomla!-Kenner befasst sich professionell sowohl mit dem Design von Websites als auch mit der Programmierung von Erweiterungen für das freie CMS. Er ist Mitgründer der deutschsprachigen SEBLOD®-Community. Seine Website time4joomla.de gilt mittlerweile als die größte kostenlose Tutorial-Website für Joomla!.

Bibliografische Information der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

Alle Angaben in diesem Buch wurden vom Autor mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. Der Verlag und der Autor sehen sich deshalb gezwungen, darauf hinzuweisen, dass sie weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernehmen können. Für die Mitteilung etwaiger Fehler sind Verlag und Autor jederzeit dankbar. Internetadressen oder Versionsnummern stellen den bei Redaktionsschluss verfügbaren Informationsstand dar. Verlag und Autor übernehmen keinerlei Verantwortung oder Haftung für Veränderungen, die sich aus nicht von ihnen zu vertretenden Umständen ergeben. Evtl. beigefügte oder zum Download angebotene Dateien und Informationen dienen ausschließlich der nicht gewerblichen Nutzung. Eine gewerbliche Nutzung ist nur mit Zustimmung des Lizenzinhabers möglich.

SEBLOD ist ein international eingetragenes Markenzeichen von Octopos company.

© 2015 Franzis Verlag GmbH, 85540 Haar bei München

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien. Das Erstellen und Verbreiten von Kopien auf Papier, auf Datenträgern oder im Internet, insbesondere als PDF, ist nur mit ausdrücklicher Genehmigung des Verlags gestattet und wird widrigenfalls strafrechtlich verfolgt.

Die meisten Produktbezeichnungen von Hard- und Software sowie Firmennamen und Firmenlogos, die in diesem Werk genannt werden, sind in der Regel gleichzeitig auch eingetragene Warenzeichen und sollten als solche betrachtet werden. Der Verlag folgt bei den Produktbezeichnungen im Wesentlichen den Schreibweisen der Hersteller.

Autor: Axel Tüting

Programmleitung: Dr. Markus Stäuble

Satz und Übersetzung: DTP-Satz A. Kugge, München

art & design: www.ideehoch2.de

ISBN 978-3-645-20470-5

Kein Vorwort

Vorworte werden nicht gelesen. Vermutlich wegen der ganzen Dankesworte, die dort stehen und niemanden interessieren. Dabei stehen zumindest in meinen Vorworten auch immer die wichtigen Dinge: für wen dieses Buch geeignet ist, und für wen nicht. Oder welche Voraussetzungen es gibt, um mit diesem Buch effektiv arbeiten zu können. Dass die allerwenigsten das lesen, sieht man spätestens in den Kritiken auf Amazon & Co., wo man sich darüber beschwert, dass man nicht die Voraussetzungen für das Buch hätte und dass man das ja irgendwo hätte hinschreiben können. Und dann schaue ich zur Sicherheit noch einmal ins Vorwort. Aber lassen wir das, verzichten wir auf das ungelesene Vorwort und kommen wir gleich zur Sache.

Wer von Joomla! keine Ahnung hat, kann mit diesem Buch nichts anfangen. Wer glaubt, er müsse nicht tiefer einsteigen und es reiche, Joomla! mal eben zu installieren, um dann tolle Sachen mit SEBLOD machen zu können, kann mit diesem Buch nicht allzu viel anfangen. Wer Joomla! zumindest rudimentär kennt und weiß, wie man Beiträge anlegen und Menüs machen kann, der kann mit SEBLOD mehr machen und das eine oder andere herausholen, was bisher nur mit Programmierkenntnissen möglich war. Wer sich mit Joomla! richtig gut auskennt, wird mit SEBLOD und diesem Buch Dinge machen können, die mit Joomla! bislang außerhalb der Möglichkeiten lagen.

SEBLOD ist eines der wenigen *Content Construction Kits* (CCK), die diesen Namen wirklich verdienen. Mehr noch: mit SEBLOD können Dinge erstellt werden, die bislang nur den Programmierern von mehr oder weniger aufwendigen Erweiterungen vorbehalten waren. Mit SEBLOD ist es oftmals möglich, in deutlich weniger Zeit komplexe Erweiterungen zu erstellen als mit der Programmierung.

Aber alles hat seinen Preis. SEBLOD ist sehr komplex und demzufolge auch nicht ganz so einfach zu durchschauen. Ein erstes Formular mit SEBLOD zu entwickeln wird hingegen verhältnismäßig schnell gehen. Komplexe Apps zu erstellen braucht seine Zeit, bis die vielen Einstellungsmöglichkeiten durchschaut und angewendet werden können. SEBLOD-Kenner sprechen davon, dass ca. 70 % aller Erweiterungen in Joomla! nicht mehr benötigt werden, da sie mit SEBLOD abbildbar sind.

Wer sich bereits mit SEBLOD befasst hat, wird aber auch wissen, dass Einstiegsliteratur mittlerweile in allen Sprachen recht verbreitet ist, weiterführende Literatur ist hingegen spärlich bis gar nicht vorhanden. Somit kann man dieses Buch durchaus auch als Pionierarbeit betrachten. Hier werden viele Dinge genannt, die nirgendwo niedergeschrieben sind und nur durch Ausprobieren und Erfahrungsaustausch mit anderen gefunden werden konnten.

Die Arbeit an diesem Buch hat lange gedauert. Sehr lange. Auch ich bin irgendwann ein Opfer der spärlichen Dokumentation geworden. Die Hilfe des französischen

SEBLOD-Teams war stets da, aber Terminkalender und meine mangelnde Kenntnis der englischen Sprache forderten auch ihren Tribut. Manchmal habe ich sogar russische Videos mit kyrillischer Schrift studiert in der Hoffnung, dass mir Klickwege Ziele verraten, zumindest ein paar mögliche Ideen geben könnten.

Auch wenn dies kein Vorwort ist, so gehören doch ein paar Dankesworte an uner-müdlische Helfer, Zuhörer und Ideengeber hinein.

Da ist zuallererst natürlich das SEBLOD-Entwicklerteam – das Team *Octopos* – zu nennen und dort mein Ansprechpartner Sebastien Lapoux.

Ein ganz besonderer Dank gebührt Paul Brand, der sich an der sonnigen Küste Spaniens mit den Tiefen von SEBLOD befasst und nicht nur die deutsche Übersetzung geschrieben hat, sondern auch ein immenses Wissen über SEBLOD hat und mich ein wenig daran teilhaben ließ.

Screendesign Whykiki kennen die einen oder anderen sicherlich von seinen tollen Videos, nicht nur zu SEBLOD. Mit bürgerlichem Namen heißt er Christian Schülling und hat ein Gastkapitel in diesem Buch bekommen. Sie finden es ab Seite 207.

Christian Linsner, der so etwas wie ein Stammtestleser geworden ist. Was dieser Mensch alles findet, ist enorm! Nie wieder ein Buch ohne ihn!

Neco Papakci von Pixelkultur hat sich durch etliche Kapitel gequält und nicht mit Anmerkungen, Tipps und Tricks gegeizt.

Dr. Markus Stäuble vom Franzis Verlag gebührt ein ganz besonderer Dank. Seine Geduld wurde auf eine harte Probe gestellt, und dennoch hat er stets an mir festgehalten und mich unterstützt!

Inhaltsverzeichnis

Teil I – Einstieg in SEBLOD	13
1 Was ist SEBLOD?.....	15
1.1 Was ist ein CCK?.....	15
1.2 Was ist mit SEBLOD möglich?	16
1.3 Versionen	17
2 Der schnelle Überblick.....	19
2.1 Installation.....	19
2.2 Apps downloaden und installieren.....	20
2.3 SEBLOD spricht deutsch.....	21
2.4 Grundsätzliche Funktionsweise	21
2.4.1 App-Folder	22
2.4.2 Formulare und Inhaltstypen.....	23
2.4.3 Felder.....	23
2.4.4 Listen & Suchtypen.....	23
2.4.5 Templates.....	24
3 Der detaillierte Blick.....	27
3.1 Die erste App: Newsanzeige	28
3.1.1 Einfache Anlage im App-Verzeichnis.....	29
3.1.2 Direkt zu den Feldern.....	32
3.1.3 Die Felder im Überblick.....	41
3.1.4 Eine Form muss her.....	41
3.1.5 Ein- und Ausgabe einer News.....	50
3.2 Die News erweitern.....	52
3.2.1 Haupteintrag und Kategorie	52
3.2.2 Bilder sagen mehr als tausend Worte	54
3.2.3 Mehr Bilder mit Feld X.....	57
3.2.4 Abhängige Bilder	59
4 Die Speicherung und ihre Hürden.....	61
4.1 Ein Blick hinter die Kulissen: MySQL	61
4.2 Speicherformat: Standard	63
4.3 Speicherformat: Benutzerdefiniert	65
4.4 Speicherformat: Json	67
4.5 Speicherformat: Kein	68
4.6 Speicherposition.....	68
4.7 Datentyp wechseln.....	70
4.8 Eigene Tabelle erstellen	70
4.9 CSV-Import.....	73

Teil 2 - Die Schaltzentrale	77
5 App-Optionen.....	79
6 ACL in der APP.....	81
6.1 Der ACL-Generalschlüssel	83
7 Optionen.....	85
7.1 Allgemein.....	85
7.2 Komponenten.....	88
7.2.1 App-Folder	88
7.2.2 Konstruktion	88
7.2.3 Entwicklung.....	89
7.2.4 Versionen.....	89
7.3 Integration	91
7.4 Sprache.....	94
7.5 Media.....	95
7.6 Multisite.....	95
7.7 Berechtigungen.....	96
7.8 Seite.....	96
7.8.1 Site (Frontend).....	96
7.8.2 Scripts & Styles.....	98
8 Alle Felder	99
8.1 JavaScript und CSS.....	100
8.2 Auswahl	101
8.2.1 Dynamisch.....	101
8.2.2 Einfach	104
8.2.3 Mehrfach.....	105
8.2.4 Zahlen	106
8.3 Button	106
8.3.1 Senden	106
8.4 Captcha.....	110
8.4.1 Mathematisch	110
8.5 Formular.....	110
8.5.1 E-Mail.....	110
8.5.2 Checkbox	110
8.5.3 Passwort.....	113
8.5.4 Radiobutton.....	113
8.5.5 Text.....	113
8.5.6 Versteckt.....	113
8.6 HTML.....	113
8.6.1 div	113
8.6.2 Icon	114
8.6.3 Tabs	114
8.7 Inhalt	117
8.7.1 Freier Text.....	117
8.7.2 Link.....	117
8.8 Joomla-Bibliothek (JForm)	118
8.8.1 Assoziationen.....	118

8.8.2	Benutzer	119
8.8.3	Benutzergruppen	119
8.8.4	Editoren	119
8.8.5	Elterngruppe.....	119
8.8.6	Hilfeseiten.....	119
8.8.7	Inhaltssprache	119
8.8.8	JForm - Kategorie	120
8.8.9	JForm - Zugangslevel	120
8.8.10	Komponenten Layout	121
8.8.11	Medien.....	121
8.8.12	Menüeintrag	122
8.8.13	Regeln.....	122
8.8.14	Schlagwörter.....	122
8.8.15	Template: Stil.....	123
8.8.16	Zeitzone.....	123
8.9	Joomla!	123
8.9.1	Artikel (Ähnliche)	124
8.9.2	Module.....	125
8.10	Kollektion/Sammlung	127
8.10.1	Gruppe	127
8.10.2	Feld X.....	127
8.10.3	Gruppe X.....	128
8.11	Picker.....	129
8.11.1	Farbauswahl.....	129
8.11.2	Kalender	129
8.12	Textarea	134
8.12.1	Textbereich	134
8.12.2	WYSIWYG-Editor.....	134
8.13	Upload	135
8.13.1	Datei	135
8.13.2	Image.....	137
8.14	Video.....	137
8.14.1	YouTube.....	137
8.15	# Core.....	138
8.15.1	42.....	138
9	Formulare & Inhaltstypen.....	141
9.1	Konfiguration: Admin- und Seitenformular.....	141
9.1.1	Konfiguration.....	141
9.1.2	Datensicherheit.....	142
9.1.3	Allgemein.....	142
9.1.4	Kein Zugang.....	143
9.1.5	Variante.....	143
9.1.6	Validation.....	143
9.2	Konfiguration: Intro und Inhalt	143
9.3	Grundeinstellungen	144
9.4	Konstruktionsbereich.....	146
9.5	CSS-Formatierung	149
9.6	Menüverweis.....	150

10 Listen & Suchtypen.....	151
10.1 Eine neue Suche	151
10.1.1 Template: Blog	152
10.1.2 Template: Table.....	153
10.1.3 Template: Kein	155
10.2 Konfiguration.....	155
10.2.1 Bereich: Konfiguration	155
10.2.2 Bereich: Allgemein.....	157
10.2.3 Bereich: Kein Ergebnis	159
10.3 Fields.....	160
10.4 Das Feld CCK	160
10.5 Nummer 3 im Suchformular.....	160
10.5.1 Basis.....	161
10.5.2 Datum und Uhrzeit	161
10.5.3 Ausschluss	161
10.5.4 Null.....	162
10.5.5 Numerisch	162
10.5.6 Wörter	162
10.6 Ordering.....	163
10.7 Liste	163
10.8 Element	163
10.9 Menüverweis.....	163
11 Die Nummern in den Formularen & Inhaltstypen.....	165
11.1 Nummer 1.....	166
11.1.1 Einzelne Felder	166
11.1.2 Admin- und Seitenformular	166
11.2 Nummer 2	167
11.2.1 Admin- und Seitenformular – Live und Live-Wert.....	168
11.2.2 Intro und Inhalt.....	170
11.3 Nummer 3	170
11.3.1 Admin- und Seitenformular	170
11.3.2 Intro und Inhalt.....	174
11.4 Nummer 4.....	176
11.5 Nummer 5	176
11.5.1 Voneinander abhängige Zustände	177
11.5.2 Rechenregeln	180
11.6 Nummer 6	181
11.7 Der Punkt.....	182
12 Das eigene Adminmenü	183
12.1 Add-ons	184
12.2 Benutzerspezifisch.....	184
12.3 Konstruktion	185
12.4 E-Commerce	186
12.5 Formulare.....	186
12.6 Lists.....	187
12.7 Fazit zum Administrationsmenü.....	187

Teil 3 – Der praktische Einsatz	189
13 Veranstaltungskalender.....	191
13.1 Die Städte.....	192
13.2 Veranstalter	193
13.3 Das Event.....	194
14 Portfolio	197
14.1 Gestaltung.....	202
15 Beiträge nach eigenem Gusto.....	205
16 Formulare für jeden Zweck	207
16.1 Warum ein Kontaktformular?.....	207
16.1.1 Die Vorbereitung: Einrichtung	208
16.1.2 Erstellen der Mailfunktion	213
16.1.3 Feldtyp Email	213
16.1.4 Absender des Formulars festlegen.....	215
16.1.5 Das Feintuning	216
16.1.6 Das Erstellen der E-Mail-Nachricht	218
16.1.7 Der Inhalt im Detail	220
16.1.8 Darstellung des Formulars	222
16.1.9 Berechtigungen	222
16.2 Erweiterung zum Beschwerdeformular	223
16.2.1 Grundlagen des Beschwerdeformulars.....	224
16.2.2 Speichern oder nicht?	227
16.2.3 Der Grundaufbau des Formulars.....	227
16.2.4 Abhängigkeiten von Feldern.....	228
16.2.5 Abhängigkeiten im Detail	228
16.2.6 Das Beschwerdeformular fertigstellen.....	231
17 Beziehungslisten	233
17.1 Beziehungs-Weise.....	235
18 Vorbelegung der Sprache bei Mehrsprachigkeit	237
19 Neue und andere Module	241
19.1 Modul: From - SEBLOD	242
19.2 Modul: List - SEBLOD	242
19.3 Modul: Search - SEBLOD	243
19.4 Modul: Breadcrumbs - SEBLOD	244
19.4.1 Ein Easteregg?	244
19.5 Module ändern.....	245
20 Multisites mit SEBLOD	247
20.1 Einrichtung einer neuen Multisite	248
20.2 Die Multi-Berechtigungen.....	252
20.3 Post von der Mastersite.....	255
20.4 Eigene Felder	255
20.5 Mastersite?	257

21	Templates	259
21.1	Grundsätzlicher Aufbau.....	259
21.2	Ein neues Template zuweisen	261
21.3	Ein eigenes Template.....	262
21.4	Overrides im seb_one.....	263
21.4.1	Positionen	263
21.4.2	Variationen.....	267
21.4.3	Aufgeräumter Quellcode mit eigenen Markups.....	270
21.4.4	Verschiedene Markup-Overrides	272
21.4.5	Zusammenfassung und Fazit	273
21.5	Einstieg mit einem einfachen Template.....	273
21.5.1	Festlegen der Positionen.....	274
21.5.2	Damit überhaupt etwas ausgegeben wird: config.php.....	275
21.5.3	Die eigentliche Ausgabe: index.php.....	276
21.5.4	Rendering-Variationen	277
21.5.5	Der Ordnung halber: die eigene CSS-Datei.....	278
21.6	Globale Methoden und Eigenschaften.....	279
22	Plug-ins	281
22.1	Plug-in-Übersicht.....	281
22.2	Wo sind die Plug-ins geblieben?	284
22.3	Verbindung zwischen Benutzer und Beitrag.....	284
A	CSS-Klassen	285
A.1	Eingabe direkt im Feld	285
A.2	Eingabe im Markup	288
B	Troubleshooting	291
B.1	WYSIWYG-Editor im Frontend	291
B.2	Merkwürdige Zeichen im Frontend	292
B.3	SQL-Fehlerbeschreibungen im Frontend	292
B.4	Fehlermeldung: Übergeordnete Kategorie	292
B.5	Das Bild im Frontend ist nicht anklickbar.....	292
B.6	Template existiert nicht	293
B.7	Anzahl der sichtbaren Felder beschränkt?	293
B.8	Nach dem Joomla!- und SEBLOD-Update	293
B.9	Mehrere Startseitenmenüs bei Multisites	294
B.10	Inhaltstyp wird nicht angezeigt	294
C	Weiterführendes	295
C.1	Deutschsprachige SEBLOD-Community	295
C.2	Tutorials	296
C.3	Videos.....	296
C.4	SEBLOD-Erweiterungen.....	296
	Stichwortverzeichnis	297

Teil I - Einstieg in SEBLOD

In diesem ersten Teil finden Sie mehr als nur eine bloße Übersicht über die SEBLOD-Struktur. Ein erstes praktisches Beispiel wird erstellt, in dem die Ein- und Ausgabe eines Joomla!-Beitrags mit SEBLOD gezeigt wird. Der erste Teil lässt bereits erkennen, wo die Stärken von SEBLOD liegen und wie einfach im Grunde die Arbeit damit ist.

Sie finden in diesem ersten Teil auch alles rund um die Speicherung. Die hat es durchaus in sich und kann zu ungeahnten Fehlern führen. Der Einstieg ist also mehr als ein bloßer Einstieg. Teilweise wird hier ordentlich hinter die Kulissen geschaut und so der Grundstock für die weitere Arbeit mit SEBLOD gelegt.



Was ist SEBLOD?

SEBLOD ist ein Content Construction Kit, kurz CCK, mit dem Dinge erstellt werden, die bislang nur Entwicklern vorbehalten waren. So lassen sich komplexe Erweiterungen mit wenig Aufwand und damit auch deutlich weniger Zeit erstellen als mit der Programmierung.

1.1 Was ist ein CCK?

CCK steht für *Content Construction Kit* und bedeutet, dass die Ein- und Ausgabemasiken für den Inhalt selbst gestaltet werden können, sowohl die Anordnung oder Anzeige von Feldern als auch das Erstellen neuer Felder, die nach Belieben eingefügt werden können. Manche CCKs fassen Inhalte sehr eng und beziehen sich meist auf die Gestaltung oder Erweiterung von Beiträgen.

Eine beliebige Erweiterung unter den Komponenten besteht stets aus Formularen, in denen etwas eingegeben werden kann oder in denen bereits etwas steht. Und im Frontend wird letztlich dieses Formular bzw. dessen Inhalt ausgegeben. Eigentlich würde es naheliegen, den Begriff »Inhalt« auch auf Erweiterungen auszudehnen. Das machen allerdings nur sehr wenige CCKs. Die wenigen, die das können, sind denn auch nicht so einfach zu durchschauen. Schon deswegen nicht, weil sie tatsächlich sehr viele unterschiedliche Bereiche und Einstellungen abdecken müssen. SEBLOD kann natürlich nicht auf bereits vorhandene externe Erweiterungen zugreifen, aber es

ist durchaus möglich, sich mit SEBLOD eigene Erweiterungen zu kreieren, ohne dass Programmierkenntnisse erforderlich sind.

Joomla! kann sehr viel. Dort, wo Joomla! von Haus aus nicht weiterkommt, helfen die über 10.000 Erweiterungen für Joomla!. Dennoch stößt man ab und an auf Grenzen oder Umständlichkeiten, zumal Erweiterungen oft genug den Nachteil haben, dass entweder ein entscheidendes kleines Detail fehlt oder aber viel zu viel dabei ist, was eigentlich gar nicht benötigt wird. Oft sind sie auch umständlich zu bedienen. Wer professionelle Webseiten erstellt, wird stets bemüht sein, seinen Kunden einen einfachen Zugang zu geben.

Selber Erweiterungen zu programmieren könnte eine Lösung sein. Aber nicht jeder kann programmieren, und eigene Erweiterungen in Joomla! zu programmieren ist nicht ganz so einfach, denn man braucht eine weitere Erweiterung – und Zeit. Ein CCK kann die Lösung für solche Probleme sein, aber auch ein CCK hat Grenzen. Es gibt Erweiterungen, die sich nicht mit einem CCK umsetzen lassen oder für die ein CCK einfach nicht geeignet ist.

SEBLOD ist eines dieser großen und mächtigen CCKs. Experten gehen davon aus, mit dem Einsatz von SEBLOD auf ca. 70 % aller Erweiterungen verzichten zu können. Jede Erweiterung, die Sie installieren, benötigt nicht nur Festplatten- und Datenbank-speicherplatz, sondern belastet auch die Performance Ihrer Website. Gerade bei größeren Websites könnte SEBLOD da eine interessante Alternative sein.

1.2 Was ist mit SEBLOD möglich?

SEBLOD wird gern als Eier legende Wollmilchsau bezeichnet, weil es eine Vielzahl von Möglichkeiten bietet und in seiner Funktionalität kaum eingeschränkt ist. Im Gegensatz zu den meisten anderen CCKs stülpt SEBLOD Joomla! nicht etwas über, sondern integriert sich nahezu komplett in Joomla!. Der Vorteil ist, dass alles, was neu in Joomla! angelegt wird, ein Beitrag, eine Kategorie, eine Benutzergruppe oder ein neuer Benutzer ist. Das sind zumindest die vorgegebenen SEBLOD-Möglichkeiten.

Durch diese Integration ist auch weiterhin alles möglich, was Joomla! von Haus aus bietet. Beispielsweise kann die Rechteverwaltung oder die Navigation komplett auf das CCK wirken, ohne dass das CCK hier selber etwas mitbringen müsste. Da also alles sehr gut weiterverwendet werden kann und lediglich etwas Neues dazukommt, muss sich aber auch niemand Gedanken darüber machen, wie etwas zur Anzeige gebracht werden kann. Ein einfacher Menüverweis reicht in aller Regel.

Ich selber habe mir ein Portfolio mit SEBLOD erstellt. Ich war mit den vorhandenen Portfoliokomponenten nicht zufrieden und hatte meine eigenen Vorstellungen, wie meine Referenzliste aussehen sollte. Das, was ich schließlich mit SEBLOD entwickelt habe, hat mich ungefähr einen halben Tag Arbeit gekostet – einschließlich umfangreicher CSS-Formatierungen.

Hätte ich das Gleiche als Komponente programmiert, hätte ich vermutlich ein paar Tage länger benötigt. Und ich habe jetzt den zusätzlichen Vorteil, dass ich jederzeit mein Portfolio umbauen oder erweitern kann, mal eben schnell ein neues Feld hinzufügen kann, ohne dass ich danach erst umständlich ein Update meiner Komponente einspielen müsste. Auch brauche ich mir keine Gedanken über neue Joomla!-Versionen und die Anpassung meiner mit SEBLOD erstellten Erweiterungen zu machen. Es reicht, neben Joomla! auch SEBLOD upzudaten, und alles ist sauber und sicher.

Da SEBLOD es ebenfalls gestattet, eigene Datenbanktabellen zu benutzen, sind in SEBLOD nur wenige Grenzen gesetzt. Ein Großteil der gängigen Joomla!-Erweiterungen ließe sich relativ problemlos in SEBLOD abbilden. Kompliziert wird es erst, wenn umfangreiche Verwaltungsaufgaben entstehen. Beispielsweise ein komplexeres Mailing- oder PM-System ist meines Erachtens in einer eigenständigen Komponente nach wie vor besser aufgehoben.

SEBLOD-Forum

Auf den Seiten von SEBLOD selber ist beispielsweise auch das Forum mit SEBLOD gemacht. Generell funktioniert das zwar, jedoch ist das Forum meines Erachtens sehr unübersichtlich, und es ist auch nicht so einfach, dort zu finden, was zusammengehört (siehe URL www.seblod.com/community/forums).

Übrigens - das deutschsprachige SEBLOD-Forum finden Sie unter <http://seblod.time4mambo.de>.

Es gibt in Joomla! einige mächtige Erweiterungen, die über die Jahre entstanden sind, stetig weiterentwickelt werden und längst von diversen Kinderkrankheiten befreit sind. Ich würde stets auf diese Erweiterungen zurückgreifen und nicht mit SEBLOD versuchen, das Rad neu zu erfinden. Aber dort, wo diese Erweiterungen nicht reichen, ist SEBLOD stets eine Überlegung wert.

Dennoch lassen sich in SEBLOD nicht nur kleine Lösungen bequem umsetzen. Es gibt beispielsweise eine SEBLOD-Erweiterung für eine Immobilienseite, die sehr umfangreich ist, sowie Dokumentationen und Download-Seiten, die mit SEBLOD erstellt sind. Ausgeklügelte Redaktionssysteme, Dokumentationen, Kundenerfahrungsberichte, Affiliateseiten in Selbstverwaltung, mehrsprachige Kataloge und Eventseiten sind nur ein paar Beispiele dafür, was mit SEBLOD möglich ist.

Dazu kommt, dass es mit SEBLOD möglich ist, Joomla! multidomainfähig zu machen. Ein Joomla! und viele Domains und Subdomains greifen auf eine »Mastersite« zu. Bislang war das nur mit (teils kostenpflichtigen) umfangreichen Erweiterungen möglich - nun ist das auch recht einfach mit SEBLOD möglich.

1.3 Versionen

SEBLOD selber gibt es bereits seit Joomla! 1.5. Die erste SEBLOD-Version hieß jSEBLOD und änderte mit der SEBLOD-Version 2 ihren Namen zu SEBLOD, die auf

Joomla! 2.5x lief. Mit Joomla! 3x kam SEBLOD 3 auf den Markt, die teilweise auch in Joomla! 2.5x noch lief. Meist wird SEBLOD zeitnah zu neuen Joomla!-Versionen ebenfalls upgedatet. Ab Joomla! 3.2 gilt der gleiche Versionsstrang von SEBLOD. Die jeweils aktuelle SEBLOD-Version finden Sie auf der SEBLOD-Site¹.

Joomla! 3.x und SEBLOD 3.x

Während dieses Buch entstand, wechselte Joomla! seine Versionspolitik. Das hatte zur Folge, dass während der Erstellung dieses Buches verschiedene Versionen zum Einsatz kamen. Entscheidend für dieses Buch ist die Joomla!-Version 3.3.6. Alles was Sie hier lesen, läuft auf dieser Joomla!-Version. Lassen Sie sich also bitte nicht von einem Screenshot erschrecken, der eine für Sie alte Versionsnummer trägt – er hat dennoch weiterhin seine Gültigkeit.

Da SEBLOD in aller Regel erst dann eingesetzt wird, wenn in Joomla! an Grenzen gerüttelt wird, setze ich voraus, dass Sie zumindest rudimentäre Kenntnisse in Joomla! haben. Auf die Handhabung und die Funktionalitäten in Joomla! gehe ich in diesem Buch nicht ein, sofern es der Umgang mit SEBLOD nicht erforderlich macht.

¹ <http://www.seblod.com>

2

Der schnelle Überblick

2.1 Installation

Die Installation von SEBLÖD ist sehr einfach. Wie in Joomla! gewohnt, wird das gezippte SEBLÖD-Package direkt über *Erweiterungen: Installieren* in Joomla! integriert.

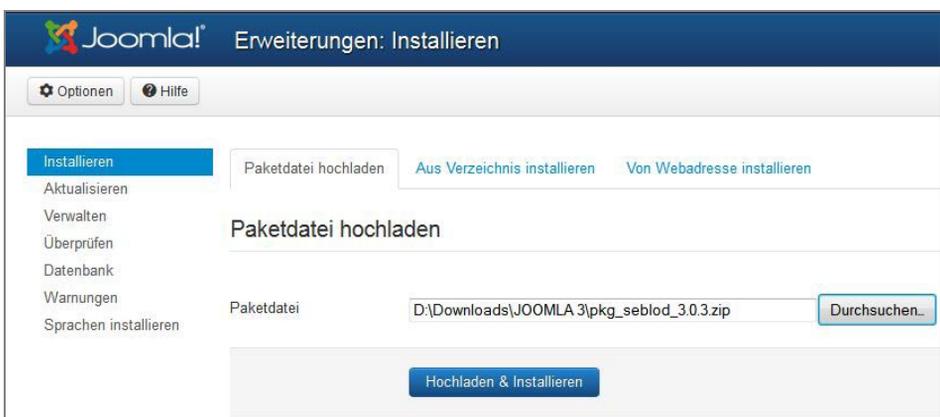


Bild 2.1: Die Installation läuft nach dem gewohnten Schema ab.

Der Installationsvorgang dauert ein wenig. Je nach Internetverbindung kann das schon einmal zwei, drei Minuten Zeit in Anspruch nehmen. Dafür legt SEBLOD 19 neue Datenbanktabellen an und installiert einige neue Dateien. In seltenen Fällen kann es dabei auch zu einem Time-out kommen.

- Eine mögliche Abhilfe besteht darin, dass Sie den Link des Downloads auf SEBLOD.com kopieren und ihn direkt im Register *Von Webadresse installieren* unter *Erweiterungen: Installieren* eingeben.
- Oder Sie laden das Paket per FTP in das *tmp*-Verzeichnis und installieren es vom Register *Aus Verzeichnis installieren* aus.
- Sollte das auch keine Abhilfe bringen, gibt es eventuell noch die Möglichkeit, unter *Erweiterungen: Installieren* auf der linken Seite die *Warnungen* auszuwählen und über die angebotenen *Optionen* (Schaltfläche oben links) den *Aktualisierungszwischenspeicher* nach oben zu setzen.

2.2 Apps downloaden und installieren

Wenn Sie fertige Apps installieren möchten (vom SEBLOD-Team werden beispielsweise einige fertige Apps angeboten), installieren Sie sie ebenfalls über *Erweiterungen: Installieren*, also so, wie Sie es kennen.

Ihre eigenen Apps mit allen Feldern und Formularen können Sie sehr einfach downloaden, indem Sie im Apps-Verzeichnis den blauen, nach unten weisenden Pfeil anklicken.



Bild 2.2: Die eigene App downloaden.

Zum einen können Sie so sehr einfach eine Sicherheitskopie Ihrer selbst erstellten App anlegen, zum anderen können Sie so natürlich eine App auch weiterreichen. Sie haben auch die Möglichkeit, lokal in aller Ruhe eine App zu entwickeln und sie erst, wenn alles fertig und getestet ist, in Ihr Live-System zu laden.

Sie speichern mit dem Download-Button Ihre komplette App, also auch alle Einstellungen, die Sie bei den Feldern und Formularen gemacht haben, und natürlich auch alle zugehörigen Formulare und Suchmasken. Wenn Ihre App weitere Untergruppierungen hat, werden diese ebenfalls gespeichert. Wenn Sie nicht die ganze App, sondern nur Teilgruppen der App sichern möchten, klicken Sie nicht den größeren blauen Pfeil, sondern den in der Zeile stehenden kleineren Pfeil an. Damit können Sie Ihre

App beispielsweise lokal weiterentwickeln und anschließend Ihre bestehende App einfach erweitern.

Was es mit den einzelnen Gruppierungen auf sich hat, werden Sie im weiteren Verlauf des Buches erfahren.

2.3 SEBLOD spricht deutsch

SEBLOD spricht auch deutsch. Es gibt zwei voneinander unabhängige Sprachversionen. Die eine kann nach einigem Suchen im SEBLOD-Forum gefunden werden und wird von Pierre Corell² entwickelt. Leider stagniert die Sprachversion seit Monaten (Stand: Oktober 2014).

Des Weiteren gibt es eine wirklich sehr gute und sehr umfangreiche Übersetzung von Paul Brand³. Paul hat nicht nur die Sprachdateien von SEBLOD, sondern auch die allermeisten Plug-ins in die deutsche Sprache übersetzt. Zudem lässt sich die jeweils aktuelle deutsche Sprachdatei einfach über das Kontrollzentrum in Joomla! aktualisieren.

In diesem Buch finden sich Screenshots von beiden Übersetzungen. Das liegt daran, dass Paul erst etwas später mit der Übersetzung begonnen hat.

2.4 Grundsätzliche Funktionsweise

Als ich das erste Mal SEBLOD installiert hatte, habe ich einige Male mehr oder weniger unkontrolliert hin- und hergeklickt und dann das Ganze wieder deinstalliert, weil ich nicht wirklich durchgestiegen bin. Tatsächlich präsentiert sich SEBLOD zunächst nicht unbedingt innovativ und auf den ersten Blick etwas unübersichtlich. Beginnt man aber, etwas Ein- und Überblick zu bekommen, wird die Struktur stetig nicht nur übersichtlicher, sondern auch logisch.

Das Faszinierendste ist für mich dabei die Erkenntnis gewesen, dass SEBLOD im Grunde genommen stets der Joomla!-Mimik folgt, also nicht etwas gänzlich Eigenes macht und Joomla! überstülpt, sondern mit dem Joomla!-System selber arbeitet und somit eine problemlose Zusammenarbeit der hauseigenen Funktionen von Joomla! mit und innerhalb von SEBLOD ermöglicht.

² <http://joomla-praxis.de> oder <http://pierre-corell.net/>

³ <http://seblod.inetserv.net>

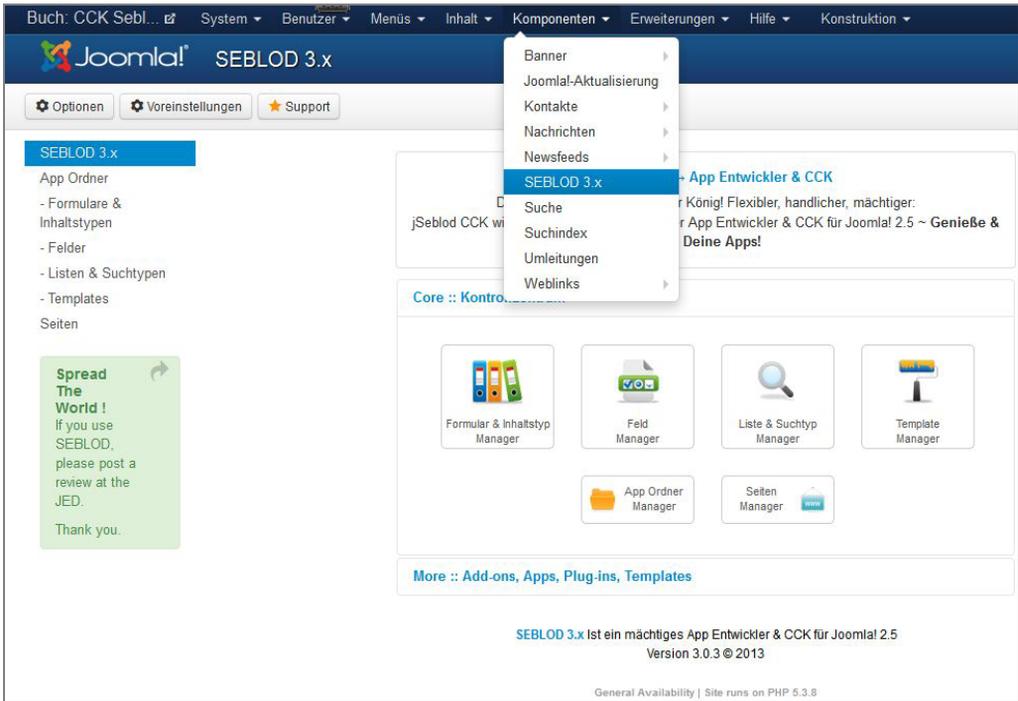


Bild 2.3: SEBLOD 3.x im Überblick.

SEBLOD legt neue Beiträge an.

Auf diesen lapidaren Satz könnte ich SEBLOD beinahe reduzieren. Bleiben wir zunächst bei dieser vereinfachten Feststellung, dann bemerken wir, dass wir zwar jede Menge eigener Erweiterungen mit SEBLOD machen können, die Verwaltung dieser eigenen Inhalte aber am Ende nichts weiter als Beiträge sind. Beiträge wiederum können auf vielfältige Weise per Menü angesprochen und gestaltet werden. Diverse Module greifen direkt auf Beiträge zu, die Suche sowieso, und die Rechteverwaltung von Joomla! weiß einiges mit Beiträgen und davon ableitend der Menüführung anzufangen.

Dadurch benötigt SEBLOD keine eigene Rechteverwaltung, weil SEBLOD einfach die vorhandene benutzt – ebenso wie die Navigation und vieles andere mehr.

SEBLOD ist komplett in Joomla! integriert. Um es durchschaubarer zu machen, werfe ich einen schnellen Blick in die einzelnen Bereiche.

2.4.1 App-Folder

»App« ist eigentlich etwas irreführend oder zumindest recht verwirrend, weil jeder Apps im Zusammenhang mit mobilen Endgeräten kennt und der Programmierer aus

der »App« noch die »Applikation« zaubert. Ich persönlich neige dazu, den »App-Folder« eher mit »Struktur« gleichzusetzen, der Struktur unserer eigenen Erweiterung in SEBLOD, zu der dann Felder und Formulare gehören. Andererseits lässt aber das Wortkürzel »App« auch erkennen, was SEBLOD genau macht: eine eigene Applikation, eine eigene Erweiterung.

Im einfachsten Fall besteht unsere SEBLOD-Erweiterung aus einem einzigen Bereich, also einer sehr klaren und ausgesprochen überschaubaren Struktur.

Weiter unten in diesem Buch bauen wir uns eine einfache News-App zusammen. Diese Newserweiterung wird aus einem Titel, etwas Text, einem Bild und einem Link bestehen. Im App-Folder bietet sich als Bezeichnung *News* an. Die vier Felder werden wir dieser Struktur zuweisen und am Ende dafür sorgen, dass das auch vernünftig ausgegeben wird.

Zusammenfassend: Der App-Folder gibt die grobe Struktur unserer SEBLOD-Erweiterung vor. Fügen wir Felder und Formulare hinzu, entsteht am Ende eine App.

2.4.2 Formulare und Inhaltstypen

An dieser Stelle sorgen Sie endlich dafür, dass es auch Formulare zum Ein- und Ausgeben von Daten gibt. Sie legen selbst fest, ob Inhalte nur im Backend vom Admin oder auch von berechtigten Benutzern im Frontend eingegeben werden können. Sie entscheiden, wie Ihre Inhalte ausgegeben werden – sowohl was das Layout als auch den Inhalt und Form betrifft. So steht ein Intro zur Verfügung für den schnellen inhaltlichen Überblick und die detaillierte Sicht. Wie genau Sie was definieren, bleibt selbstverständlich vollkommen Ihnen überlassen.

2.4.3 Felder

Wie bereits erwähnt, werden in diesem Register die Felder unserer App erstellt. In meinem Beispiel mit den News erstelle ich die vier Felder *Titel*, *Text*, *Bild* und *Verweis* und weise sie meinem App-Folder zu.

Möglich ist an dieser Stelle natürlich auch, statt einer eigenen App eine bereits vorhandene zu erweitern, beispielsweise die Joomla!-Beiträge, wie wir sie bislang kennen. Sie können zwar auch Felder direkt bei den Formularen und Inhaltstypen zuweisen, aber hier sind die Übersicht und die Handhabung etwas besser.

2.4.4 Listen & Suchtypen

Ich kann die News in meinem Beispiel auflisten, indem ich nur den Titel anzeige, oder ich kann die Auflistung als Blog mit dem Text und dem Verweis darstellen oder – wie Sie es möchten. Hier bestimmen Sie, wie solche Listen aussehen sollen.

Und auch die Suche können Sie selber bestimmen. Da Sie auch Abhängigkeiten definieren können – beispielsweise *Dieses Feld wird erst angezeigt, wenn in dem Feld dieses und jenes steht* –, können Sie bereits erahnen, dass Sie vielfältige Suchmuster erschaffen können. Sie entscheiden, was gesucht und wie es ausgegeben wird.

2.4.5 Templates

Das hat mich anfangs sehr verwirrt: Unter Templates versteht man in Joomla! stets Design und Layout einer Website. Tatsächlich finden sich auch die mitgelieferten SEBLOD-Templates in der Template-Übersicht wieder – allerdings mit dem strikten Hinweis, dass sie sich nicht zur Seitenanzeige eignen. Probieren Sie es dennoch aus, passiert nichts Sinnvolles auf Ihrer Webseite.

Template und Template

SEBLOD unterscheidet zwischen – Template und Template. Falls Sie gerade die Augen verdrehen sollten – ich bringe gleich Licht in diese Merkwürdigkeit.

Wenn Sie einen Beitrag in Joomla! eingeben, dann sehen Sie oben zunächst den Titel, daneben die Kategorieauswahl, darunter den Editor, darunter unter anderem die Bilder und Links, rechts neben dem Editor ein paar Details zum Auswählen und weitere Tabs mit verschiedenen Einstellungen. Dieser ganze Bereich wird in SEBLOD Template genannt.

Zudem gibt es das Template nur und ausschließlich für die Eingabe des Beitrags. Dessen Aussehen können Sie umgestalten, indem Sie bestimmen, an welcher Stelle was abgebildet wird, indem Sie die einzelnen Felder einfach anderen Positionen zuordnen oder indem Sie ein ganz neues eigenes Template entwickeln.

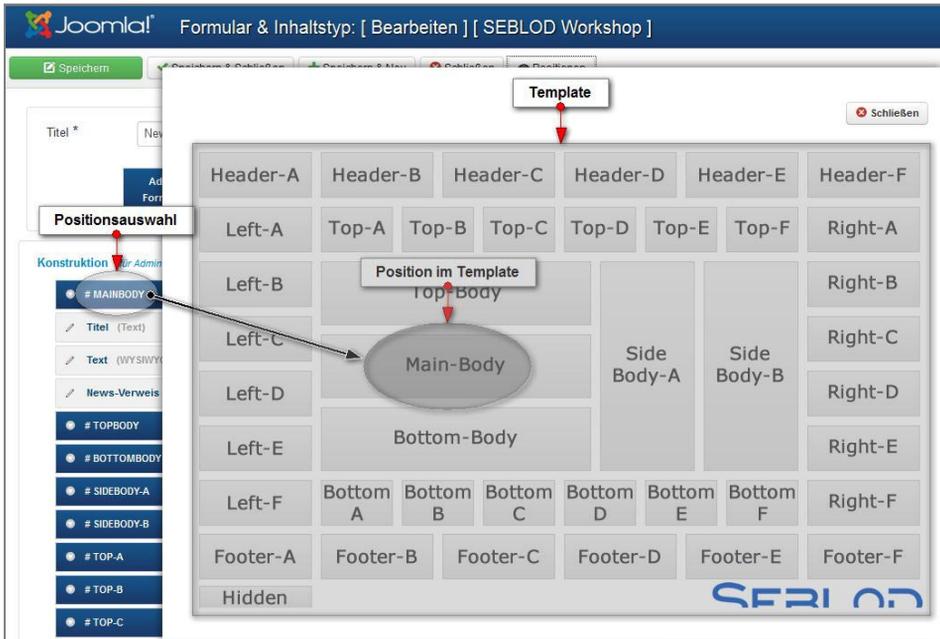


Bild 2.4: Template-Positionen in SEBLOD.

Zusammengefasst bedeutet das: Das SEBLOD-Template umfasst nur den Inhaltsbereich, nicht das Gesamtaussehen der Website. Dieser ganze Ablauf ist am Ende ein einzelner Beitrag, bestehend aus Struktur - Feldern - Form und gegebenenfalls eigenen Auflistungen und Suchen.

Was ich eben erklärt habe, gilt ebenso für Kategorien, die Sie umgestalten können, wie auch für Benutzer und Benutzergruppen, die Sie so nach Ihrem eigenen Gusto anlegen können.

App (Beiträge, Kategorien, Benutzer, Benutzergruppen)



Und alles läuft stets wie gehabt innerhalb von Joomla!. Mit SEBLOD erstellen Sie sich zwar Ihre eigenen Ein- und Ausgabemasken, handhaben anschließend aber alles im normalen Joomla!-Kontext.

Meine News-App (die im nächsten Kapitel angelegt wird) sieht zwar komplett anders aus als ein Joomla!-Beitrag, wie wir ihn bisher kennen, aber ich kann diese selbst erstellten News ganz normal, und wie wir es bisher kennen, im Menü ansprechen – als Kategorieblog und Liste, als Einzelverlinkung oder als Archiv. Und ebenso kann ich mit der Rechteverwaltung auf diese Beiträge zugreifen, und mir stehen somit alle Joomla!-Möglichkeiten offen.

Bei anderen CCKs haben Sie hingegen meist das Problem, dass Sie das dort Erstellte nur so verarbeiten können, wie es das jeweilige CCK vorgibt. Die gewonnene Freiheit des CCK wird oft mit Einschränkungen der Weiterverarbeitung wieder zunichtegemacht. Nicht so in SEBLOD.

Teil 3 - Der praktische Einsatz

Was mit SEBLOD möglich ist und wie man es vielfältig einsetzen kann, ist immer noch am besten erklärt, wenn man es in der Praxis anwendet. In diesem dritten Teil geht es mir weniger um fertige Lösungen, die Sie nur noch herunterladen und zu installieren brauchen, um dann fertige Erweiterungen zu haben, sondern es geht mir darum, anhand einfacher Beispiele einige Möglichkeiten aufzuzeigen. Sie werden aber ohne nennenswerte Schwierigkeiten die Beispiele an Ihre Bedürfnisse anpassen können.



Veranstaltungskalender

Die Überlegung ist, dass ich mehrere Städte habe, in diesen Städten Veranstaltungsorte sind und Künstler dort auftreten. Meine Website soll über die Künstler berichten, weshalb die Künstlerbeiträge als Joomla!-Beiträge abgespeichert werden. Sowohl die Stadtauswahl als auch die einzelnen Veranstalter werden lediglich in der Datenbank gespeichert, sollen aber als Auswahllisten in den Künstlerbeiträgen zur Verfügung stehen.

Dazu benötige ich drei Inhaltstypen:

- Eingabe der Städte, die anschließend dynamisch ausgewählt werden.
- Eingabe der Veranstalter, die bei den Events ebenfalls dynamisch ausgewählt werden.
- Das Event.

Alles kommt in die App *Veranstaltungen*.

	Titel	App Ordner	# Admin Formular	# Seitenformular	# Intro	# Inhalt
	Event event	 Veranstaltungen TOP	3	-	2	3
	Veranstalter veranstaltungsort	 Veranstaltungen TOP	3	-	-	-
	Veranstaltungsstädte eingeben veranstaltungen	 Veranstaltungen TOP	1	-	-	-

Bild 13.1: Drei Inhaltstypen für die *Veranstaltungen*.

In der Abbildung sehen Sie, dass ich lediglich dem Inhaltstyp *Event* Ausgaben im Frontend gebe, die beiden anderen Inhaltstypen bekommen nur ein Adminformular.

13.1 Die Städte

Die Städte möchte ich als Auswahlliste bei den Veranstaltern haben. Später will ich eine Suchliste erstellen und mir so alle Veranstaltungsorte in einer Stadt anzeigen lassen. Dazu erstelle ich zuerst eine neue Tabelle in meiner Datenbank. Dabei gehe ich so vor, wie in Kapitel 4.8 »Eigene Tabelle erstellen« beschrieben. Neben der ID lege ich eine Spalte mit dem Namen *stadt* an und nenne die Tabelle *#__veranstaltungen*.

Nun wird ein neuer Inhaltstyp angelegt, der App *Veranstaltungen* zugeordnet und ein Feld von Typ *Text* im *Admin Formular* angelegt. Im Speicherbereich wähle ich zunächst *Standard* aus und dann *Frei*, damit ich meine gerade angelegte Datenbanktabelle und die Tabellenspalte *stadt* auswählen kann.

Speicher ²

Format / Position Standard ▾ Frei ▾ fj51_veranstaltungen ▾

stadt « Wechseln

Bild 13.2: Speicherung des Textfeldes.

Das war es bereits, um die Städte einzugeben. Wenn Sie nun in die Beiträge gehen und dieses Formular aufrufen, sehen Sie ein unscheinbares Eingabefeld. Völlig unspektakulär. Dadurch, dass wir aber in einer eigenen Tabelle speichern, erscheinen die einzelnen Städte nicht in der Beitragsübersicht.

13.2 Veranstalter

Für die Veranstalter lege ich wieder eine neue Datenbanktabelle an, diesmal aber neben der ID mit drei weiteren Spalten:

- *id*
- *stadt*
- *veranstaltungsname*
- *anschrift*



+ Optionen				id	veranstaltungsname	stadt	anschrift
<input type="checkbox"/>		Bearbeiten		Kopieren		Löschen	10 Pavillon Hannover Weiße-Kreuz-Platz
<input type="checkbox"/>		Bearbeiten		Kopieren		Löschen	11 Oper Hannover Operplatz
<input type="checkbox"/>		Bearbeiten		Kopieren		Löschen	12 Theater am Aegi Hannover Aegidientorplatz
<input type="checkbox"/>		Bearbeiten		Kopieren		Löschen	13 Tollhaus Karlsruhe

Bild 13.3: Die Datenbanktabelle mit einigen Einträgen.

Die drei weiteren Spalten sind alle als *Text* angelegt. Genannt habe ich die Tabelle *#_veranstalter*.

Im *Admin Formular* erstelle ich nun drei neue Felder. *Name* und *Anschrift* sind einfache Textfelder, wobei Sie die *Anschrift* auch als *Textarea* anlegen können. Alle drei Felder speichern Sie als *Standard* und *Frei* in die neu angelegte Tabelle.

Die *Stadt* lege ich vom *Typ: Dynamisch* an. Damit kann ich auf die Datenbanktabelle mit den Städten zugreifen und diese als Auswahlliste anzeigen.

Lassen Sie die *Query* auf *Konstruktion* stehen und tragen Sie in *Table: #_veranstaltungen* ein – genau so, wie ich es hier geschrieben habe, also mit *#_* (dem doppelten Unterstrich). In *Optionsname* kommt der Spaltenname, dessen Inhalte Sie angezeigt bekommen möchten, und in *Optionswert*, welcher Inhalt abgespeichert werden soll. In unserem Fall ist das sehr einfach, weil wir nur eine Tabellenspalte haben, die wir gebrauchen können: die Spalte mit den Städten.

Gemacht ist das so, weil es ja gut sein könnte, dass Sie neben dem Stadtnamen auch noch die PLZ haben. Dann können Sie die Stadt beispielsweise per Auswahlliste anzeigen, die zugehörige PLZ aber beim Veranstalter speichern. Näheres zu diesem Feldtyp erhalten Sie im Kapitel 8 »Alle Felder«. In unserem Fall kommt also in beiden Feldern *stadt* hinein.

Alternativ können Sie auch bei *Query* auf *Frei* umschalten und die SQL-Syntax direkt eingeben:

```
001 SELECT stadt FROM fj51l_veranstaltungen
```

Dann müssen Sie aber das Präfix Ihrer Datenbanktabellen mit angeben. In der *Konstruktion* brauchen Sie das nicht.

Wenn Sie nun in der Beitragsübersicht einen neuen Veranstalter eingeben wollen, erhalten Sie eine dynamische Auswahlliste der bisher eingetragenen Städte.

Bild 13.4: Dynamische Auswahllisten.

Auch diese Einträge werden nicht als Joomla!-Beiträge gespeichert, weil wir sie in einer eigenen Datenbanktabelle abgespeichert haben.

13.3 Das Event

Das eigentliche Event wird nun endlich unser Beitrag, den wir entsprechend ausgeben können. In diesem kleinen Beispiel mache ich es mir einfach und lege lediglich den Künstler und das Auftrittsdatum an. Den Veranstaltungsort hole ich mir wieder als *dynamisches Feld*, allerdings diesmal von der Datenbanktabelle `#_veranstalter`. Der *Optionsname* ist der *veranstaltername*, und diesmal speichere ich die *id* ab, die ich im *Optionenwert* eintrage.

Query	Konstruktion ▾	Wähle Bezeichnung	Wähle einen Veranstaltungsort
Table	#_veranstalter	Optionsname	veranstaltername
Wo		Optionenwert	id

Bild 13.5: Die *id* wird abgespeichert.

Das mache ich deswegen, weil ich in der Ausgabe nicht nur auf den Veranstalternamen, sondern auch auf die anderen Spalten zugreifen möchte.

Veranstaltungsort und *Datum* speichere ich jeweils *benutzerdefiniert* im *introtxt* des *Artikels*. Den *Künstlernamen* speichere ich als *Standard* im *Artikel* und in der Spalte *title* ab. Damit kann ich meine einzelnen Events optimal in der Beitragsübersicht wiederfinden. Das *Intro* und den *Inhalt* gestalten Sie nach eigenem Gusto.

Rufen Sie nun einen Beitrag im Frontend auf, dann erhalten Sie zwar die ID des Veranstaltungsorts, aber mehr leider nicht. Um Stadt, Anschrift und Name des Veranstaltungsorts anzuzeigen, benötigen wir ein Plug-in, das leider nicht standardmäßig bei SEBLOD dabei ist. Aber unter der Product-Site²⁹ auf SEBLOD gibt es das Plug-in *SD Databaser* zum kostenlosen Download.

Mit dem *SD Databaser* können Sie Ausgaben von Datenbanktabellenzellen machen, auf die Ihr Beitrag zwar verweist, die aber nicht direkt ausgegeben werden können. Also genau das, was wir vorliegen haben: Wir haben eine ID gespeichert, die auf einen Datensatz verweist, den wir aber nicht direkt anzeigen können.

Das Plug-in installieren Sie so, wie in Joomla! üblich – also über *Erweiterungen/Erweiterungen/Paketinstallation*. Eventuell müssen Sie das Plug-in in der Plug-in-Übersicht von Joomla! noch aktivieren. Wechseln Sie nun zu Ihrem *Formular & Inhaltstyp* und rufen Sie den *Inhalt* Ihres Eventformulars auf (wahlweise das *Intro*). Klicken Sie hier die *Nummer 3* an. Damit wechseln Sie in die *Typografie*. Bei dem Feld für den Veranstaltungsort wählen Sie in der Spalte *Typografie* ganz unten *SD Databaser* aus. Sie erhalten nun ein kleines Pluszeichen, das Sie bitte anklicken.



Bild 13.6: Auswahl des *SD Databaser*.

Das Plug-in ist eines der wenigen Plug-ins auf SEBLOD, das wirklich sehr gut dokumentiert ist. Das Video beim Download dieses Plug-ins zeigt einige Möglichkeiten auf. Bei mir sieht leider das Plug-in nicht so hübsch aus wie auf dem Video, erfüllt aber dennoch seinen Zweck.

Wir benötigen den Veranstalternamen, die Stadt des Veranstaltungsorts und die Anschrift. Somit müssen wir zuerst eine SQL-Anweisung formulieren und in das Feld *SQL Query* eingeben. In unserem Fall lautet sie wie folgt:

```
001 select veranstaltername, stadt, anschrift from #__veranstalter where
    id=*value*
```

²⁹ <http://www.seblod.com/products/2239>

Sie können hier mit #__ arbeiten. Die Datenbanktabelle hatten wir zuvor selber angelegt. Übergeben müssen wir die ID, die wir gespeichert haben. Das tun wir mit der Variablen `value`, die in Sternchen eingefasst werden muss. Wenn Sie das speichern und ins Frontend zurückkehren, werden Sie leider immer noch unsere abgespeicherte ID sehen. Aber in *Column(s) AS Value* tragen wir nun endlich ein, welches oder welche Felder ausgegeben werden sollen, in unserem Fall also die drei Felder, die auch in der SQL-Abfrage bereits genannt sind. Allerdings müssen wir die drei Felder mit zwei senkrechten Strichen trennen (| |). Sie kommen an die senkrechten Striche, indem Sie auf der Windows-Tastatur `AltGr` + `←` drücken. Der korrekte Eintrag lautet:

```
001 Veranstaltername||stadt||anschrift
```

Damit die einzelnen Daten untereinanderstehen, tragen Sie in *Column(s) Separator* einen HTML-Zeilenumbruch ein:

```
001 <br/>
```

Multiple Row Separator benötigen Sie, wenn Sie mehrere Datensätze ausgeben und die einzelnen Datensätze optisch voneinander trennen möchten. Beispielsweise könnten Sie unsere Veranstalterdaten auch mit einem Komma trennen, indem Sie als *Column(s) Separator* ein Komma mit einem nachfolgenden Leerzeichen eintragen und in *Multiple Row Separator* den HTML-Zeilenumbruch. Zum Testen ändern Sie die SQL-Abfrage am Ende auf `id >= 1`:

```
001 select Veranstaltername, stadt, anschrift from #__veranstalter where
id>=1
```

Vorausgesetzt, Sie haben mindestens zwei Veranstalter eingetragen, werden diese auch entsprechend ausgegeben.

Der zerbrochene

Veranstaltungsort 1

Veranstaltungsort 2

Veranstaltungsort

Pavillon, Hannover, Weiße-Kreuz-Platz

Oper, Hannover, Opernplatz

Theater am Aegi, Hannover, Aegidientorplatz

Tollhaus, Karlsruhe, und so weiter...

Hotel Ehrwürdig, Braunschweig, Platz der Großen

Künstler: Der zerbrochene Krug

Auftrittsdatum: 2013-11-29

Kategorie: Apfelkuchen

Bild 13.7: Mehrere Veranstaltungsorte mit dem *SD Databaser* ausgeben.

Fazit: Ein kleines Beispiel, wie Sie mit mehreren Tabellen Ihre Daten sauber trennen und am Ende wieder zusammenfügen können. Einziger Nachteil bei dieser Methode: Sie müssen sich ein wenig mit SQL vertraut machen. SQL selber ist aber einfach zu erlernen, und für unsere Belange reicht Ihnen das Grundwissen, um alles Benötigte umsetzen zu können.

Stichwortverzeichnis

Symbole

#_ 34
#_categories 68
#_cck_store_form_ 68
#_cck_store_form_article 75
#_cck_store_item_ 68
#_cck_store_item_users 68
#_content 34, 68
#_usergroups 68
#_users 68
#name_des_feldes# 220
#SIDEBODY-A 47
\$cck 276, 279
\$cck->id 280
\$option->get 269
:: 65
:cck:: 284
_blank 40

A

Abgerundete Ecken 126
Abhängigkeit 176
 Checkboxen 238
ACL 81, 83, 96
Add Button 91
Add-ons 73, 184
addStyleSheet 278
Admin Formular 43, 46, 54
Administrationsmenü 184
Adminmenü 183
Adminmodule 184

Ajax 122, 256
 Availability 171
Aktion 160
Alias 33, 65, 213
Aliasnamen 99
alt-Attribut 38
Alternativtext 38
App 23, 192
App Optionen 79
App Ordner 210
App Ordner Core 260
App Prozeß 80
App Root (1. Ebene) 30
App-Folder 23, 99
Apps installieren 20
Archiviert 83
Article Category Id 233, 292
Article-App 54
Article-Title 234
Artikel 294
Artikel (Ähnliche) 234
Artikelnummer 158
Associations 93
Ausgabe der Formate 85
Auslöser 59
AUTO_INCREMENT 72
Automatische Speicherung 85
Automatische Weiterleitung 157
Autor 74

B

Barrierefreiheit 38

Bearbeiten 83
Bearbeitungssymbol 97
Beez 3 29
Benutzer 25
Benutzerdefiniert 65
Benutzergruppen 25, 119
Benutzericon 251
Berechtigungen 81, 96, 122, 145, 223, 251
Beschreibung 99
Beschreibungstexte 48
Bezeichnung 35, 99
Bild 174
blank 109
Blog 152
Bootstrap 285
BOTTOM 48
Braillegeräte 38
Brand, Paul 6, 21, 296
Breadcrumbs 241, 244
Brotkrumen (s. Breadcrumbs)
Brücke 284
Buttons 68

C

Cache 155
Category 53
Category Parent Id 292
category zebra table 154
catid 242
CCK 15, 160
CCK 5
CCK Feld - Wysiwyg Editor 291

- cck_storage_location 284
 - change mode 136
 - chmod 136
 - clear 48
 - com_cck_ecommerce 186
 - config.php 276
 - Content Construction Kid
 - 5, 15
 - Core 69, 70
 - Core-Elemente 205
 - Corell, Pierre 21
 - CSS 285
 - CSS-Dateien 259
 - CSS-Klassen 40, 100, 181
 - CSV-Datei 73
 - CSV-Daten 63
 - CSV-Format 63
- D**
- Daten 62
 - Datenbank 62, 192
 - Datenbankabfragen 62
 - Datenbanktabelle 20, 62, 193
 - Datenfelder 67
 - Datensätze 62
 - Datentyp 70
 - Datenzelle 62
 - Datetime 70
 - de-DE.com_cck.ini 95
 - Default Inhaltstyp 91
 - dfContact 208
 - Disable List Template 155
 - Dokumentation 28
 - Doppelte Doppelpunkte 66
 - Downloaden 79
 - Duplicate entry 260
 - Dynamisch 193
- E**
- Easteregg 244
 - E-Commerce 186
 - Eigene Attribute 100
 - Eigene Inhalte bearbeiten 83
 - Eigene Tabelle 192
 - Eigentümer 96
 - Einfach (Simple) 212
 - Einleitungen 152
 - Elemente 31, 152, 163, 208
 - Beschriftung 158
 - Email 213
 - E-Mails 207
 - embed 138
 - Endgültige Sortierung 243
 - Entwicklungs-Präfix 88
 - Entwicklungsspeicher 89
 - Entwicklungs-Speicherung 256
 - Erlaubt 82
 - Erlaubte Erweiterungen 36
 - Erstellen 82
 - Erweiterter Modus 38
 - Erweiterung 15
 - Erzeuge Tabelle 71
 - Excel-Tabelle 63, 73
 - Exporter 107
- F**
- Featured 52
 - Fehlermeldungen 159
 - Fehlersuche 156
 - Feld (Formular) 57
 - Feld CCK 160
 - Feld X 127
 - Felder 44, 153
 - Feldtyp 70
 - Bild 35
 - Feld X 57
 - Kategorie 52
 - Link 39
 - Medien 54
 - Text 33
 - Textbereich 39
 - WYSIWYG-Editor 39
- G**
- FlexiContact 208
 - Forciere 92
 - Format 63, 69
 - Format/Präcision 180
 - Formulare 15, 23
 - Formulare & Inhaltstypen 43
 - Framesets 109
 - Frei 68
 - Führender 152
 - fulltext 65
 - Fußzeile 48
- H**
- h1 55, 157
 - h2 55, 158

Hamberger, Christian 296
Hauptbeitrag 52
Haupteintrag 31
HEADER 48, 56
Homepage 294
HTML 65, 211

I

id 235
ID 62, 72
Iframe 138
Image 36, 174
Image-Feldtyp 292
images 38
index.html 262
Informationen 62
Inhalt 15, 43, 46
 erstellen 156
 vorbereiten 88
Inhaltserstellung 224
Inhaltssprache 93
Inhaltstyp 76, 151
Inkludiere Seiten-Name
 250
Integration 91
Intro 43, 46
introtext 41
ItemID 241

J

Jardin, David 296
JavaScript 67, 98, 100
JForm 118
JForms 54
Joomla User - [ID] 284
Joomla! 65
 multidomainfähig 17
Joomla!-Bibliothek 52
jSEBLOD 17
Json 67, 257
JText 85, 94

K

Kanonische Seite 97
Kategorie 74
Kategorieblog 26, 152,
 201, 235
Kategorieliste 151, 152, 153
Kategorien 25
Kein 68
Kein Ergebnis 159
Kiki 207
Kollation 72
Komponenten 15
konfiguriere 237
Konstruktion 35
Kopfzeile 48, 56

L

Label 99, 210
Landing Page 113
lang 237
Leere ACL 83
LEFT 48
Limit 155
Linke Spalte (Breite) 47
Links 152
Linsner, Christian 6
Liste 151, 153, 158, 163
Listen 23
Listen & Suchtypen 151,
 187

Live Wert 234
loadposition 242
localhost 71
Löschen 83

M

MAINBODY 43, 46
mainbody.php 265
markup 269, 279
Markup 288
markup.php 269

max_input_vars 293
Maximalgröße 35
Media 95
Mehrfachkategorie-
 auswahl 120
Mehrsprachigkeit 118, 237
Menü Konstruktion 28
Meta 251
Microsoft-Server 61
MindMap 226
Mindmeister 226
Miniaturbild 38, 55
Mitteilungen 159
Mittel 100, 285
Modalbox 175
Modul 125, 234, 235
moduletable 125
Modulposition 235, 263
More Link 242
Multidomain 96
Multidomainfähig 17
Multilanguage 93, 118
Multisites 89, 294
MySQL 61
MySQL-Datenbank 61
MySQL-Dokumentation
 70

N

Nachrichten 159
Nachrichtenstil 141, 159
Navigation 22
Neco Papakci 6
Nested 123
News-App 23
Nübel, Niels 296
Null 162
Nummer 45, 160
Nummer 2 237
Nummer 5 201
Nutze Allgemein 156

O

Objekt 294
 Octopos 6
 Offline 258
 Onlineshop 139
 option.xml 268
 Optionale Variante 156
 Optionen 47, 85, 256
 Felder 256
 Multisites 257
 Ordering 163
 Override 121
 (Live/Variation) 164

P

pagination 159
 Panel 114
 Paniermehl 244
 Papierkorb 83
 parent 40, 109
 parent_id 119
 Pflichtfeld 33
 PHP 67
 php.ini 293
 phpMyAdmin 71
 Pixelkultur 6
 Placebo 160
 Plesk 248
 PLG_CCK_FIELD_GROUP
 282
 Plug-in
 Bestätigung 281
 Download 283
 Erweiterte Optionen
 281
 Export 283
 Gruppe 282
 Site (Multi-sites): 256
 Speicherung 281
 System - Sprachenfilter
 93
 Typografie 281

 Validation 281
 Plus 55
 Portfolio 16
 Positionen 24, 46, 65, 153
 PrepareContent 139
 PrepareForm 138
 PrepareStore 139
 Prozesse 35, 36
 Pull-down-Felder 242

Q

Query 193

R

Random 243
 Raw Rendering 164
 Rechte Spalte (Breite) 47
 Rechtesystem 248
 Rechteverwaltung 22, 81
 Reihenfolge 163
 Relativer Pfad 279
 renderField 280
 rendering.php 275
 Rendering-Klasse 276
 Responsives Webdesign
 97
 Revision 90

S

Save \$ Redirect 107
 Save & Skip 107
 Schlagwörter 122
 Schnelle Ordner 261
 Schnellstartmenü 183
 Schülling, Christian 6, 296
 Screendesign Whykiki 6
 SD Databaser 195
 Sebastien Lapoux 6
 SEBLOD 5
 SEBLOD-Version 17
 SEF 156

Seitenanzahl 156
 Seitenformular 43, 46
 Seitennummerierung 159
 Seitenteil-a 47
 Seitenzahl 156
 Selbst 156
 self 40, 109
 SEO 156, 235
 Shopsystem 186
 Shortcut 185
 Spalte anhängen 70
 Spalten 62, 152
 Spaltennamen 63
 Speicherort Entwicklung
 257
 Speicherposition 68
 Speicherung 292
 Sprachen 86
 Sprachschlüssel 94, 282
 Sprachversionen 21
 SQL 61, 102, 193
 SQL-Fehler 292
 Standard Autor 86
 Standard Reihenfolge 157
 Standardeditor 135
 Standardsortierung 243
 Standard-Wert 33
 Statische Optionen 103
 Status 30, 59, 74
 Status bearbeiten 83
 Struktur 23
 Suche 24, 151
 Suchergebnisse 153
 Suchformular 153, 155
 Suchmaschinen-
 optimierung 158
 Symbol 31
 Systemmeldungen 159

T

Tabelle 62, 154

- Tabellenstruktur 69
 - Table 152, 153
 - Table Layout 154
 - target 40
 - Task 107
 - Template 24, 153, 154
 - Blog 260
 - client 274
 - Computation 270
 - Conditional 270
 - content 264
 - desc 270
 - fieldset 268
 - files 275
 - html 270
 - index.php 276
 - Inhalt 264
 - Inhaltstyp 272
 - Kein 152
 - label 270
 - method 274
 - Override 262
 - Positionen 263
 - Suchtyp 272
 - tbody.php 264
 - Varianten 259
 - Variation 267
 - Template Style does not exist 293
 - template_preview.png 273
 - template_thumbnail.png 273
 - templateDetails.xml 274
 - Textbereich 210
 - Textfeld 134
 - Time-out 20
 - title 33
 - Toolbox-Add-on 107
 - top 40, 109
 - Tr (Gerade) 154
 - Trenner 74
 - Trigger 229
 - Trigger Event 180
 - Typografie 55, 174
 - Typografie Bild 292
- U**
- Überschrift 157
 - UNIQUE 72
 - Unter Bezeichnung 153
 - Unter dem Feld 153
 - Unter Formular/Wert 153
 - Update 69, 293
 - URL Assignment 241
 - URL Query (Vars) 150, 164
 - URL zusammensetzen 100
 - utf8_general_ci 72
- V**
- Variable 237
 - Verborgen 233
 - Vererbt 82
 - Veröffentlicht 83
 - Version vor der Wiederherstellung 90
 - Versionen 89
 - Verweigert 82
 - Verzeichnisrechte 136
 - Voreinstellungen 85
- W**
- Weiterlesen 94
 - Whykiki 207, 296
 - Wiederherstellung 90
 - WYSIWYG-Editor 30, 291
- X**
- XML 268
- Y**
- YouTube 137, 207
- Z**
- Zeichen 31
 - Zeichencodierung 72
 - Zeige Beschreibung 158
 - Zeige Link 40
 - Zeige Löschen 35
 - Zeige Seitenzahl 159
 - Zeige spezifischen Pfad 37
 - Zeige Text 40
 - Zeige Titel 157
 - Zeige Vorschau 36
 - Zeige Ziel 40
 - Zeilenumbruch 65
 - Zugriffsebenen 120

SEBLOD® Handbuch

Das Content-Management-System Joomla! stellt dem Anwender bereits eine breite Palette von Funktionen zur Ausgestaltung von Webauftritten bereit. Bei erfolgreichen Webseiten steigt mit der Zeit aber der Anspruch in Bezug auf Funktionalität. Es ist nur eine Frage der Zeit, bis Sie an die Grenzen von Joomla! stoßen. Entweder greifen Sie dann in den Programmcode ein oder setzen auf ein Content Construction Kit (CCK). Das mächtigste dieser Art ist SEBLOD®. Wie Sie damit die Grenzen von Joomla! sprengen, zeigt Ihnen das vorliegende Handbuch.

Einstieg mit einem Praxisbeispiel

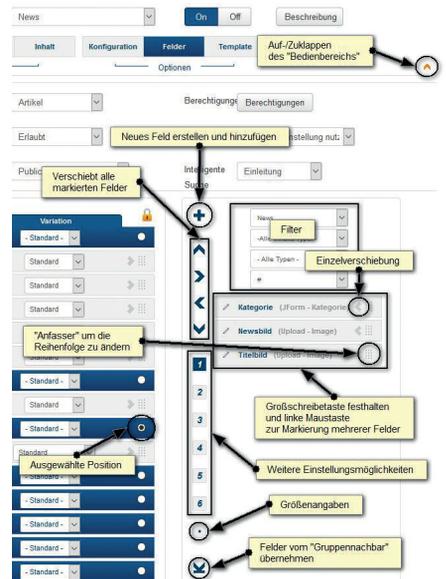
Das CCK SEBLOD® bietet eine große Fülle an Funktionen, die es zu kennen gilt. Nach der Installation und einer Einführung in die Grundlagen erlernen Sie an einem ersten Beispiel die Arbeitsweise mit SEBLOD®. Schritt für Schritt lernen Sie dabei die Entwicklung einer eigenen SEBLOD®-App, wie die mit SEBLOD® erstellten Erweiterungen genannt werden. Am Beispiel einer Newsanzeige setzen Sie u. a. Felder und Formulare ein.

Detailwissen für große Erweiterungen

Die Einstellmöglichkeiten in SEBLOD® sind schier unerschöpflich. Im zweiten Teil dieses Handbuchs werden detailliert die Rechteverwaltung, mögliche Optionen und insbesondere alle verfügbaren Felder erklärt. Auch Formulare, Inhaltstypen, Listen und Suchtypen werden ausführlich behandelt. Dem wichtigen Thema der Templates wird ein komplettes Kapitel gewidmet. Abgerundet wird das Buch mit einem Anhang, der Hilfe bei auftretenden Fehlern bietet.

Aus dem Inhalt:

- Was ist ein Content Construction Kit?
- SEBLOD® installieren und mit Apps erweitern
- Funktionsweise von SEBLOD®
- Newsanzeige-App selbst entwickeln
- Datenspeicherung in MySQL
- GSV-Import
- Rechteverwaltung
- SEBLOD®-Optionen im Detail
- Feldtypen im Überblick
- Formulare & Inhaltstypen
- Listen & Suchtypen
- Eigenes Admin-Menü
- Veranstaltungskalender
- Formulare für jeden Zweck
- Beziehungslisten
- Multisites mit SEBLOD®
- Templates
- Plug-ins



Reich bebilderte Beschreibung der Oberfläche von SEBLOD®.