

Tobias Schlitt, Kore Nordmann

eZ Components

Das Entwickler-Handbuch

Inhalt

Geleitwort des Fachgutachters	13
Vorwort	17

1 Einleitung 21

1.1 eZ Systems	21
1.1.1 Wie alles begann	21
1.1.2 Das Geschäftsmodell	22
1.2 eZ Components	24
1.2.1 Wie alles begann	24
1.2.2 Was sind eZ Components?	25
1.2.3 Übersicht der Komponenten	29
1.3 Das Galileo-Press-Blog	36
1.3.1 Was soll entwickelt werden?	37
1.3.2 Aufbau der Beispielanwendung	37
1.3.3 Testen des GP-Blogs	41
1.3.4 Rewriting einrichten	42
1.4 Der weitere Verlauf	45

2 Einführung in eZ Components 49

2.1 eZ Components installieren	49
2.1.1 Versionierung	50
2.1.2 Installation per PEAR	51
2.1.3 Das Gesamtpaket herunterladen	54
2.1.4 SVN	56
2.2 Autoload konfigurieren	58
2.2.1 eZ Components Autoload einschalten	58
2.2.2 Interna des eZ Components Autoloads	60
2.2.3 Eigenes Autoload einbinden	61
2.2.4 Preloading	63
2.3 Weitere Bestandteile der Base-Komponente	64
2.3.1 Optionen-Klassen	64
2.3.2 Exceptions	73
2.3.3 ezcBaseFeatures	75
2.3.4 Lazy-Initialization	76
2.4 Dokumentation	80
2.4.1 API-Dokumentation	80
2.4.2 Anleitungen	82

- 2.4.3 Mitgeliefertes 82
- 2.4.4 Beispielapplikationen und Artikel 83
- 2.4.5 Support 83
- 2.5 Selbst etwas beisteuern 84
 - 2.5.1 Fehler melden 84
 - 2.5.2 Patches bereitstellen und Dokumentation verbessern 86
 - 2.5.3 Eigene Komponenten beisteuern 87
 - 2.5.4 Contributor-License-Agreement 88

3 Die Applikationsbasis 91

- 3.1 Das MVC Pattern 91
- 3.2 Verwendete Komponenten 92
 - 3.2.1 Url 93
 - 3.2.2 SignalSlot 93
- 3.3 Der Haupt-Controller 94
 - 3.3.1 Grundstruktur 95
 - 3.3.2 Initialisierung 96
 - 3.3.3 Signale verarbeiten 99
 - 3.3.4 Read-only-Attribute 100
 - 3.3.5 Laufen lassen 102
- 3.4 Die Action-Controller 105
 - 3.4.1 Ein einfacher Action-Controller 106
 - 3.4.2 Einträge editieren 110
- 3.5 Erweiterte Möglichkeiten 111
 - 3.5.1 Ungeordnete Parameter 111
 - 3.5.2 URLs verwalten 112
 - 3.5.3 Signale priorisieren 113
 - 3.5.4 Statische Signale 114
 - 3.5.5 Lazy-Initialization für SignalSlot 116
- 3.6 Fazit 117

4 Fehlerbehandlung und Debugging 119

- 4.1 Verwendete Komponenten 120
 - 4.1.1 Execution 120
 - 4.1.2 Debug 120
- 4.2 Fehlerbehandlung 121
 - 4.2.1 Ausgaben puffern 121
 - 4.2.2 Die Fehler-Aktion 122
 - 4.2.3 Fatale Fehler 124

4.3	Debugging	126
4.3.1	Debug-Ausgaben	126
4.3.2	Fehlerinformationen	128
4.3.3	Timing-Informationen	130
4.3.4	Debug-Ausgabe	133
4.4	Fazit	135

5 Konfiguration 137

5.1	Nutzen in der Applikation	137
5.2	Die Configuration-Komponente	138
5.3	Integration der Komponente	138
5.3.1	Basiseinstellungen	139
5.3.2	Komplexe Indizes	141
5.4	Direkter Zugriff	142
5.4.1	Änderungen speichern	143
5.4.2	Schnellerer Zugriff durch PHP-Arrays	144
5.4.3	Lazy-Initialization	146
5.5	Fazit	147

6 Datenbankbindung 149

6.1	Abstraktion	149
6.1.1	Abstraktion der API	149
6.1.2	SQL-Abstraktion	150
6.1.3	Unterstützte RDBMS	150
6.2	Nutzung in der Applikation	151
6.3	Die Database-Komponente	151
6.3.1	PDO	152
6.3.2	Fluent-Interfaces	152
6.4	Integration von Database	155
6.4.1	Ein neues Blog Posting erstellen	156
6.4.2	Prepared-Statements	159
6.4.3	Update	160
6.4.4	Daten auslesen	162
6.5	Komplexere Abfragen	164
6.5.1	Manuelle Queries	166
6.5.2	Sub-Selects	167
6.5.3	Joins	168
6.5.4	LIMIT und ORDER BY	169

6.6	Erweiterte Möglichkeiten	170
6.6.1	Lazy-Initialization	170
6.6.2	Ein anderes RDBMS verwenden	172
6.7	Fazit	173

7 ORM mit PersistentObject 175

7.1	Modell-Klassen	175
7.2	Konzepte	178
7.2.1	Active Record	178
7.2.2	Row Data Gateway	179
7.2.3	Persistent Object	180
7.3	Konfiguration	180
7.3.1	Basiskonfiguration	181
7.3.2	Relationen beschreiben	185
7.4	Benutzung	189
7.4.1	Eine Persistenz-Sitzung erzeugen	189
7.4.2	Objekte auffinden	192
7.4.3	Objekte manipulieren	195
7.5	Erweiterte Möglichkeiten	200
7.5.1	Über Abfragen manipulieren	200
7.5.2	Werte auffrischen	201
7.5.3	Direkt mit der Datenbank arbeiten	201
7.6	Fazit	201

8 Template 203

8.1	Template-Komponente	204
8.2	Cross-Site-Scripting	204
8.2.1	Ausgabe-Kontext in der Template-Komponente	206
8.3	Integration in das Blog	207
8.3.1	Initialisierung der Template-Komponente	207
8.3.2	Übergabe von Inhalten an das Template	208
8.3.3	Wichtige Variablen	209
8.3.4	Ausgabe	209
8.3.5	Template-Struktur im Blog	210
8.3.6	Das pagelayout-Template	210
8.3.7	Einbinden von Modul-Templates in die Applikation	212
8.3.8	Liste der Blog-Einträge	213
8.3.9	Vollansicht eines Blog-Eintrags	215
8.3.10	Gebietsabhängiges Datum	217

8.4	Weitere Möglichkeiten der Template-Komponente	221
8.4.1	Rückgaben verarbeiten	221
8.4.2	Eigene Template-Blöcke definieren	222
8.5	Fazit	223

9 Übersetzung 225

9.1	Komponente ezcTranslation	225
9.2	Qt Linguist	226
9.2.1	Das Dateiformat	226
9.2.2	Das Programm	227
9.3	Einbindung in die Applikation	228
9.3.1	Template-Funktion	228
9.3.2	Einfügen nicht vorhandener Übersetzungen	232
9.3.3	Templates anpassen	235
9.3.4	Parametrisierte Übersetzungen	236
9.3.5	Übersetzen	237
9.3.6	Filter	239
9.4	Caching von Übersetzungen	241
9.5	Fazit	242

10 Benutzereingaben validieren 243

10.1	Sicherheitsprobleme	243
10.1.1	Benutzereingaben in Desktop-Applikationen	244
10.1.2	Betrachtung des Webbereichs	244
10.1.3	PHP-Sicherheit	245
10.2	Lösung mit userInput	246
10.2.1	Formulare definieren	247
10.2.2	Sicherer Zugriff auf Formularvariablen	251
10.2.3	Ein weiteres Beispiel	253
10.3	Erweiterte Möglichkeiten	255
10.4	Fazit	256

11 Authentifizierung 257

11.1	Die Authentication-Komponente	258
11.2	Einbindung in die Applikation	259
11.2.1	Eingabeformulare	259
11.2.2	Formularverarbeitung anpassen	260
11.2.3	Authentifizierung des Besuchers	261

- 11.2.4 Gruppierung von Filtern 263
- 11.2.5 OpenID 264
- 11.3 Benutzer in der Datenbank 267
- 11.4 Fazit 269

12 Bildverarbeitung 271

- 12.1 Bilder analysieren 272
 - 12.1.1 Architektur 272
 - 12.1.2 Handler und Backends 276
- 12.2 Bilder manipulieren 278
 - 12.2.1 Architektur 279
 - 12.2.2 Handler und Backends 284
 - 12.2.3 Im GP-Blog 290
- 12.3 Erweiterte Möglichkeiten 300
 - 12.3.1 Exif-Daten 301
 - 12.3.2 Weitere Filter 301
 - 12.3.3 ImageAnalysis erweitern 304
 - 12.3.4 ImageConversion erweitern 305
- 12.4 Fazit 310

13 Archive und Dateien 311

- 13.1 Die Archive-Komponente 311
- 13.2 Einbindung in die Applikation 312
 - 13.2.1 Übersichtsseite 312
 - 13.2.2 Archiv erstellen 313
 - 13.2.3 Dateien an ein Archiv anhängen 316
 - 13.2.4 Archive auslesen 317
- 13.3 Die File-Komponente 320
 - 13.3.1 Rekursives Löschen 320
- 13.4 Fazit 320

14 Mail 321

- 14.1 E-Mails senden 321
 - 14.1.1 Architektur 322
 - 14.1.2 E-Mail-Versand 326
- 14.2 E-Mails empfangen 332
 - 14.2.1 Architektur 332
 - 14.2.2 E-Mail-Empfang 336

14.3	Weitere Möglichkeiten	349
14.3.1	IMAP-Flags	350
14.3.2	IMAP-Postkästen	350
14.3.3	Zeichencodierungen	351
14.3.4	Kleine Helferlein	352
14.4	Fazit	353

15 Logging 355

15.1	Die EventLog-Komponente	355
15.2	Einbindung in die Applikation	355
15.2.1	Fehler protokollieren	356
15.2.2	Weitere Nachrichten loggen	358
15.2.3	Zusätzliche Daten loggen	359
15.2.4	Kategorien und Quelle	360
15.2.5	Eigener Log-Writer	362
15.3	Weitere Möglichkeiten mit EventLog	364
15.3.1	Loggen in der Datenbank	364
15.3.2	Lazy-Initialization	365
15.4	Fazit	366

16 Diagramme 367

16.1	Die Graph-Komponente	367
16.1.1	Ausgabeformate	368
16.2	Einbindung in die Applikation	369
16.2.1	Die Aktion erstellen	369
16.2.2	Einfaches Pie-Chart	372
16.2.3	Dreidimensional	374
16.2.4	Eigene Palette definieren	377
16.2.5	Liniendiagramm erstellen	379
16.2.6	Formatierung der Elemente	381
16.3	Weitere Möglichkeiten	382
16.3.1	Generierung von Bitmaps	382
16.3.2	Balkendiagramme	384
16.4	Fazit	387

17 Feeds und Caching 389

17.1	Feed	390
17.1.1	Feeds parsen	391
17.1.2	Feeds erzeugen	392

17.2	Die Cache-Komponente	394
17.2.1	Motivation	394
17.2.2	Architektur	395
17.2.3	Praktisches	398
17.3	Cache-Attribute	401
17.4	Fazit	402

18 Setup 403

18.1	Architektur	403
18.1.1	DatabaseSchema	404
18.1.2	Die ConsoleTools-Komponente	406
18.1.3	Eingaben	410
18.1.4	Tabellen	414
18.1.5	Dialoge	415
18.1.6	Fortschritt	418
18.2	Praxis im Blog	423
18.2.1	Das aktuelle Schema auslesen	424
18.2.2	Ein Setup-Programm	429
18.3	Erweiterte Möglichkeiten	437
18.3.1	Beziehungen zwischen Optionen	437
18.3.2	Optionsaliase	439
18.4	Fazit	440

Anhang 441

A	Inhalt der Buch-CD	441
A.1	add-ons/	441
A.2	doc/	441
A.3	ezc/	441
A.4	misc/	441
A.5	stage01/ ... stage16/	442

Index	443
-------------	-----