

Andi Gutmans, Stig Sæther Bakken, Derick Rethans

PHP 5 aus erster Hand

Das Entwicklerhandbuch für Profis



An imprint of Pearson Education

München • Boston • San Francisco • Harlow, England
Don Mills, Ontario • Sydney • Mexico City
Madrid • Amsterdam

Inhaltsverzeichnis

Vorwort	29
Danksagungen	31
Einleitung	33
Der Anfang	33
PHP/FI 2	33
PHP 3	34
PHP 4	34
PHP 5	35
Leserkreis	35
Überblick über die einzelnen Kapitel	36
Ein Hinweis zum Codierungsstil	38
Über die Software	38
Aktualisierungen, Errata und Downloads	39
1 Was gibt's Neues bei PHP 5?	41
1.1 Einführung	41
1.2 Sprachfunktionen	41
1.2.1 Das neue objektorientierte Modell	41
1.2.2 Die neuen objektorientierten Funktionen	43
1.2.3 Weitere neue Sprachfunktionen	48
1.3 Allgemeine Änderungen bei PHP	49
1.3.1 XML und Webdienste	49
1.4 Weitere neue Funktionen in PHP 5	52
1.4.1 Der neue Memory-Manager	52
1.4.2 Keine Unterstützung mehr für Windows 95	52
1.5 Zusammenfassung	53
2 Grundlegendes zu PHP 5	55
2.1 Einführung	55
2.2 HTML-Einbettung	55
2.3 Kommentare	56

2.4	Variablen	57
2.4.1	Variable Variablen	58
2.4.2	Umgang mit Variablen	58
2.4.3	Superglobale Arrays	60
2.5	Grundlegende Datentypen	61
2.5.1	Ganzzahlen	61
2.5.2	Fließkommazahlen	61
2.5.3	Strings	62
2.5.4	Boole'sche Werte	64
2.5.5	NULL	66
2.5.6	Resources	66
2.5.7	Arrays	66
2.5.8	Konstanten	73
2.6	Operatoren	74
2.6.1	Binäroperatoren	75
2.6.2	Zuweisungsoperatoren	76
2.6.3	Vergleichsoperatoren	77
2.6.4	Logische Operatoren	78
2.6.5	Bitweise Operatoren	78
2.6.6	Unäroperatoren	79
2.6.7	Negationsoperatoren	79
2.6.8	Inkrement- und Dekrement-Operatoren	80
2.6.9	Cast-Operatoren	81
2.6.10	Der @-Operator	81
2.6.11	Der Ternäroperator	82
2.7	Kontrollstrukturen	82
2.7.1	Bedingte Steuerung	82
2.7.2	Steuerung von Schleifen	86
2.7.3	Steuerstrukturen zum Einschließen von Quelltext	89
2.8	Funktionen	91
2.8.1	Benutzerdefinierte Funktionen	92
2.8.2	Funktionsumfang	93
2.8.3	Werte als Wert zurückgeben	94
2.8.4	Werte als Referenz zurückgeben	94
2.8.5	Funktionsparameter deklarieren	95
2.8.6	Statische Variablen	97
2.9	Zusammenfassung	97

3	Objektorientierung in PHP	99
3.1	Einführung	99
3.2	Objekte	99
3.3	Eine Klasse deklarieren	101
3.4	Das Schlüsselwort new und Konstruktoren	101
3.5	Destruktoren	103
3.6	Zugriff auf Methoden und Eigenschaften mit der Variable \$this	104
3.6.1	public, protected und private für Eigenschaften	104
3.6.2	Methoden als public, protected und private deklarieren	106
3.6.3	Statische Eigenschaften	107
3.6.4	Statische Methoden	108
3.7	Klassen-Konstanten	109
3.8	Objekte klonen	110
3.9	Polymorphismus	112
3.10	parent:: und self::	115
3.11	Der Operator instanceof	115
3.12	Abstrakte Methoden und Klassen	117
3.13	Interfaces	118
3.14	Interfaces vererben	120
3.15	final-Schlüsselworte	120
3.16	final	121
3.17	Die Methode __toString()	122
3.18	Ausnahmebehandlung	123
3.19	__autoload()	126
3.20	Hinweise auf Klassentypen in Funktionsparametern	128
3.21	Zusammenfassung	129
4	Erweiterte OOP und Entwurfsmuster	131
4.1	Einführung	131
4.2	Fähigkeit zum Überladen	131
4.2.1	Überladen von Eigenschaften und Methoden	131
4.2.2	Die Syntax für den Arrayzugriff überladen	134
4.3	Iteratoren	135

4.4	Entwurfsmuster	140
4.4.1	Strategy-Pattern	141
4.4.2	Singleton-Pattern	143
4.4.3	Factory-Pattern	144
4.4.4	Observer-Pattern	148
4.5	Reflektion	150
4.5.1	Einführung	150
4.5.2	Reflection-API	150
4.5.3	class ReflectionException extends Exception Beispiele für Reflektion	153
4.5.4	Implementieren des Delegation-Patterns durch Reflektion	154
4.6	Zusammenfassung	156
5	Webanwendungen in PHP	157
5.1	Einführung	157
5.2	Einbetten in HTML	158
5.3	Benutzereingaben	160
5.4	Sicherer Umgang mit Benutzereingaben	163
5.4.1	Häufige Fehler	163
5.5	Techniken, die Scripts sicherer machen	166
5.5.1	Eingabeüberprüfung	166
5.5.2	HMAC-Überprüfung	168
5.5.3	PEAR::Crypt HMAC	169
5.5.4	Eingabefilter	173
5.5.5	Mit Passwörtern arbeiten	173
5.5.6	Fehlerbehandlung	175
5.6	Cookies	176
5.7	Sessions	180
5.8	Datei-Uploads	183
5.8.1	Umgehen mit der hochgeladenen Datei	184
5.9	Architektur	189
5.9.1	Ein Script für alle Fälle	189
5.9.2	Ein Script für jede Funktion	190
5.9.3	Die Logik vom Layout trennen	191
5.10	Zusammenfassung	192

6	Datenbanken mit PHP 5	195
6.1	Einführung	195
6.2	MySQL	196
6.2.1	Stärken und Schwächen von MySQL	196
6.2.2	PHP-Schnittstelle	197
6.2.3	Beispieldaten	197
6.2.4	Verbindungen	197
6.2.5	Gepufferte und ungepufferte Abfragen	200
6.2.6	Abfragen	200
6.2.7	Mehrere Anweisungen	202
6.2.8	Fetch-Modi	202
6.2.9	Vorbereitete Anweisungen	203
6.2.10	BLOBs behandeln	205
6.3	SQLite	207
6.3.1	Stärken und Schwächen von SQLite	207
6.3.2	Die besten Einsatzgebiete	208
6.3.3	Die PHP-Schnittstelle	209
6.4	PEAR DB	225
6.4.1	Wie Sie PEAR DB erhalten	226
6.4.2	Pro und Kontra der Datenbankabstraktion	226
6.4.3	Welche Funktionen werden abstrahiert?	226
6.4.4	Datenbankverbindungen	227
6.4.5	Abfragen ausführen	229
6.4.6	Ergebnisse holen	233
6.4.7	Sequenzen	235
6.4.8	Funktionen für die Portabilität	236
6.4.9	Abstrakte Fehler	238
6.4.10	Methoden, die Ihnen die Arbeit erleichtern	239
6.5	Zusammenfassung	241
7	Fehlerbehandlung	243
7.1	Einführung	243
7.2	Fehlertypen	244
7.2.1	Programmierfehler	244
7.2.2	Undefinierte Elemente	247
7.2.3	Portabilitätsfehler	249
7.2.4	Laufzeitfehler	253
7.2.5	PHP-Fehler	254

7.3	PEAR-Fehler	259
7.3.1	Die Klasse PEAR_Error	263
7.3.2	Behandeln von PEAR-Fehlern	266
7.3.3	PEAR-Fehlermodi	267
7.3.4	Vernünftiger Umgang mit Fehlern	268
7.4	Ausnahmen	271
7.4.1	Was sind Ausnahmen?	271
7.4.2	try, catch und throw	271
7.5	Zusammenfassung	273
8	XML mit PHP 5	275
8.1	Einführung	275
8.2	Das Vokabular	276
8.3	XML parsen	278
8.3.1	SAX	278
8.3.2	DOM	282
8.4	SimpleXML	287
8.4.1	Ein SimpleXML-Objekt erzeugen	287
8.4.2	SimpleXML-Objekte ansehen	288
8.4.3	SimpleXML-Objekte speichern	290
8.5	PEAR	290
8.5.1	XML_Tree	290
8.5.2	XML_RSS	291
8.6	XML konvertieren	294
8.6.1	XSLT	295
8.7	Mit XML kommunizieren	299
8.7.1	XML-RPC	300
8.7.2	SOAP	308
8.8	Zusammenfassung	315
9	Mainstream-Extensions	317
9.1	Einführung	317
9.2	Dateien und Streams	317
9.2.1	Zugriff auf Dateien	318
9.2.2	Programmeingabe/-ausgabe	320
9.2.3	Ein-/Ausgabe-Streams	323

9.2.4	Compression-Streams	325
9.2.5	User-Streams	326
9.2.6	URL-Streams	328
9.2.7	Sperren	333
9.2.8	Dateien umbenennen und entfernen	334
9.2.9	Temporäre Dateien	335
9.3	Reguläre Ausdrücke	335
9.3.1	Syntax	336
9.3.2	Funktionen	349
9.4	Datumsbehandlung	357
9.4.1	Datum und Uhrzeit abrufen	358
9.4.2	Datum und Uhrzeit formatieren	361
9.4.3	Datumsformate parsen	369
9.5	Grafikbearbeitung mit GD	371
9.5.1	Fallstudie 1: Bot-sichere Formulare	372
9.5.2	Fallstudie 2: Balkengrafik	377
9.5.3	EXIF	383
9.6	Multibyte-Strings und Zeichensätze	386
9.6.1	Zeichensätze umwandeln	387
9.6.2	Sonderfunktionen für Multibyte-Zeichensätze	392
9.6.3	Locales	396
9.7	Zusammenfassung	400
10	PEAR verwenden	401
10.1	Einführung	401
10.2	Das Konzept von PEAR	402
10.2.1	Packages	402
10.2.2	Versionen	403
10.2.3	Versionsnummern	403
10.3	Wie Sie PEAR erhalten	406
10.3.1	Installation auf einer Unix-/Linux-PHP-Distribution	406
10.3.2	Windows	407
10.3.3	go-pear.org	408
10.4	Packages installieren	412
10.4.1	Den Befehl pear verwenden	412
10.5	Konfigurationsparameter	415

10.5.1	Konfigurationsschichten	418
10.5.2	Ändern der Konfiguration	419
10.6	PEAR-Befehle	422
10.6.1	pear install	423
10.6.2	pear list	427
10.6.3	pear info	428
10.6.4	pear list-all	429
10.6.5	pear list-upgrades	430
10.6.6	pear upgrade	430
10.6.7	pear upgrade-all	431
10.6.8	pear uninstall	432
10.6.9	pear search	433
10.6.10	pear remote-list	433
10.6.11	pear remote-info	435
10.6.12	pear download	435
10.6.13	pear config-get	436
10.6.14	pear config-set	436
10.6.15	pear config-show	436
10.6.16	Abkürzungen	437
10.7	Frontends für Installationsprogramme	438
10.7.1	CLI-Installationsprogramm	438
10.7.2	Gtk-Installationsprogramm	438
10.8	Zusammenfassung	442
11	Wichtige PEAR-Packages	443
11.1	Einführung	443
11.2	Datenbank-Abfragen	443
11.3	Template-Systeme	443
11.3.1	Begriffe für Templates	444
11.3.2	HTML_Template_IT	444
11.3.3	HTML_Template_Flexy	448
11.4	Authentifizierung	453
11.4.1	Übersicht	453
11.4.2	Beispiel: Authentifizierung mit Passwortdatei	453
11.4.3	Beispiel: Authentifizierung mit Datenbank und Benutzerdaten	455
11.4.4	Sicherheitsüberlegungen zur Authentifizierung	457

11.4.5	Die Skalierbarkeit der Authentifizierung	458
11.4.6	Zusammenfassung der Authentifizierung	460
11.5	Umgang mit Formularen	460
11.5.1	HTML_QuickForm	460
11.5.2	Beispiel: Login-Formular	460
11.5.3	Daten empfangen	461
11.5.4	Caching	461
11.5.5	Cache_Lite	461
11.6	Zusammenfassung	463
12	PEAR-Komponenten erzeugen	465
12.1	Einführung	465
12.2	PEAR-Standards	465
12.2.1	Symbolbenennung	465
12.2.2	Einrückung	468
12.3	Versionsnummern	470
12.4	CLI-Umgebung	471
12.5	Grundlagen	472
12.5.1	Include-Dateien: Wann und wie einsetzen?	472
12.5.2	Fehlerbehandlung	474
12.6	Packages erzeugen	474
12.6.1	PEAR-Beispiel: Hello World	474
12.6.2	Den Tarball erzeugen	477
12.6.3	Überprüfung	477
12.6.4	Regressionstests	478
12.7	Das package.xml-Format	479
12.7.1	Package-Informationen	480
12.7.2	Versionsinformationen	482
12.8	Abhängigkeit	487
12.8.1	Element <code><deps></code>	487
12.8.2	Das Element <code><dep></code>	487
12.8.3	Abhängigkeitstypen	489
12.8.4	Warum Sie Abhängigkeiten vermeiden sollten	490
12.8.5	Optionale Abhängigkeiten	491
12.8.6	Einige Beispiele	491

12.9	Substitution von Zeichenketten	491
12.9.1	Das Element <code><replace></code>	492
12.9.2	Beispiele	492
12.10	Einschließen von C-Code	493
12.10.1	Das Element <code><configureoptions></code>	493
12.10.2	Das Element <code><configureoption></code>	493
12.11	Packages veröffentlichen	494
12.12	Der Release-Prozess von PEAR	494
12.13	Packaging	495
12.13.1	Codeanalyse	496
12.13.2	MD5-Prüfsummen generieren	496
12.13.3	Update von Package.xml	496
12.13.4	Eine Tarball-Datei erzeugen	497
12.14	Upload	497
12.14.1	Upload-Release	497
12.14.2	Fertig!	498
12.15	Zusammenfassung	498
13	Die Migration vollziehen	499
13.1	Einführung	499
13.2	Das Objektmodell	499
13.3	Objekte in Funktionen parsen	500
13.4	Kompatibilitätsmodi	501
13.4.1	Objekte umwandeln (Casting)	501
13.4.2	Objekte vergleichen	502
13.5	Weitere Änderungen	503
13.5.1	Zuweisen zu <code>\$this</code>	503
13.5.2	<code>get_class()</code>	506
13.6	<code>E_STRICT</code>	506
13.6.1	Objekte automatisch erzeugen	507
13.6.2	<code>var</code> und <code>public</code>	507
13.6.3	Konstruktoren	507
13.6.4	Vererbte Methoden	508
13.6.5	Klassen vor ihrer Verwendung definieren	509

13.7	Weitere Kompatibilitätsprobleme	509
13.7.1	Die Befehlszeile	509
13.7.2	Befehls-Token	509
13.7.3	MySQL	510
13.8	Änderungen bei Funktionen	511
13.8.1	array_merge()	511
13.8.2	strrpos() und strrpos()	512
13.9	Zusammenfassung	513
14	Performance	515
14.1	Einführung	515
14.2	Leistungsorientierter Entwurf	515
14.2.1	PHP-Entwurfstipp 1: Vorsicht vor dem Status	516
14.2.2	PHP-Entwurfstipp 2: Cache!	518
14.2.3	PHP-Entwurfstipp 3: Nicht überentwerfen!	523
14.3	Benchmarking	524
14.3.1	ApacheBench verwenden	524
14.3.2	Siege verwenden	525
14.3.3	Tests und echter Traffic	526
14.4	Der Zend Studio-Profiler	526
14.5	Profiling mit APD	528
14.5.1	APD installieren	529
14.5.2	Trace-Daten analysieren	530
14.6	Profiling mit Xdebug	533
14.6.1	Xdebug installieren	534
14.6.2	Tracing von Scripts	534
14.6.3	Kcachegrind verwenden	536
14.7	APC (Advanced PHP Cache) verwenden	539
14.8	ZPS (Zend Performance Suite) verwenden	540
14.8.1	Automatisch optimieren	540
14.8.2	Caching kompilierten Codes	541
14.8.3	Caching dynamischen Inhalts	543
14.8.4	Inhalte komprimieren	546
14.9	Den Code optimieren	547
14.9.1	Mikro-Benchmarks	547
14.9.2	Neu schreiben in C	550
14.9.3	OO und prozeduraler Code im Vergleich	551
14.10	Zusammenfassung	552

15	Einführung in das Schreiben von PHP-Erweiterungen	553
15.1	Einführung	553
15.2	Ein schneller Einstieg	554
15.2.1	Speicherverwaltung	559
15.2.2	Werte aus PHP-Funktionen zurückgeben	560
15.2.3	self_concat() fertig stellen	561
15.2.4	Zusammenfassung des Beispiels	563
15.2.5	Wrapping() der Erweiterungen von Drittanbietern	563
15.2.6	Globale Variablen	573
15.2.7	Eigene INI-Direktiven hinzufügen	574
15.2.8	Threadsichere Makros für den Ressourcen-Manager	576
15.3	Zusammenfassung	577
16	Shell-Scripts mit PHP	579
16.1	Einführung	579
16.2	CLI-Shell-Scripts mit PHP	579
16.2.1	Unterschiede zwischen CLI und CGI	580
16.2.2	Die Umgebung für das Shell-Scripting	582
16.2.3	Befehlszeilenoptionen parsen	584
16.2.4	Guter Stil	587
16.2.5	Prozesssteuerung	588
16.2.6	Beispiele	593
16.3	Zusammenfassung	598
A	Index der PEAR- und PECL-Packages	601
A.1	Authentifizierung	601
A.1.1	Auth	601
A.1.2	Auth_Enterprise	602
A.1.3	Auth_HTTP	602
A.1.4	Auth_PrefManager	602
A.1.5	Auth_RADIUS	603
A.1.6	Auth_SASL	603
A.1.7	LiveUser	603
A.1.8	radius	604
A.1.9	sasl	604
A.2	Benchmarking	605
A.2.1	Benchmark	605

A.3	Caching	605
A.3.1	APC	605
A.3.2	Cache	605
A.3.3	Cache_Lite	605
A.4	Konfiguration	606
A.4.1	Config	606
A.5	Konsole	606
A.5.1	Console_Color	606
A.5.2	Console_Getopt	607
A.5.3	Console_ProgressBar	607
A.5.4	Console_Table	607
A.5.5	ecasound	607
A.5.6	win32std	608
A.5.7	xmms	608
A.6	Datenbank	609
A.6.1	DB	609
A.6.2	DBA	609
A.6.3	DBA_Relational	610
A.6.4	dbplus	610
A.6.5	DB_ado	610
A.6.6	DB_DataObject	611
A.6.7	DB_DataObject_FormBuilder	611
A.6.8	DB_ldap	612
A.6.9	DB_ldap2	612
A.6.10	DB_NestedSet	612
A.6.11	DB_Pager	613
A.6.12	DB_QueryTool	613
A.6.13	DB_Table	613
A.6.14	Gtk_MDB_Designer	614
A.6.15	isis	614
A.6.16	MDB	615
A.6.17	MDB2	616
A.6.18	mbdtools	617
A.6.19	MDB_QueryTool	618
A.6.20	oci8	618
A.6.21	odbt	618
A.6.22	Paradox	619
A.6.23	SQLite	619
A.6.24	SQL_Parser	619

A.7	Datum und Uhrzeit	620
A.7.1	Calendar	620
A.7.2	Date	620
A.7.3	date_time	620
A.8	Verschlüsselung	621
A.8.1	Crypt_CBC	621
A.8.2	Crypt_CHAP	621
A.8.3	Crypt_Crypt	621
A.8.4	Crypt_HMAC	622
A.8.5	Crypt_RC4	622
A.8.6	Crypt_Xtea	622
A.8.7	Message	622
A.9	Dateiformate	622
A.9.1	Contact_Vcard_Build	623
A.9.2	Contact_Vcard_Parse	623
A.9.3	Fileinfo	623
A.9.4	File_DICOM	624
A.9.5	File_Fstab	624
A.9.6	File_Gettext	624
A.9.7	File_IMC	625
A.9.8	File_Ogg	625
A.9.9	Genealogy_Gedcom	625
A.9.10	MP3_ID	625
A.9.11	Spreadsheet_Excel_Writer	626
A.9.12	zip	626
A.10	Dateisystem	626
A.10.1	Archive_Tar	626
A.10.2	Archive_Zip	627
A.10.3	File	627
A.10.4	File_Find	627
A.10.5	File_HtAccess	627
A.10.6	File_Passwd	627
A.10.7	File_SearchReplace	628
A.10.8	File_SMBPasswd	628
A.10.9	VFS	628
A.11	Gtk-Komponenten	629
A.11.1	Gtk_VarDump	629

A.12	HTML	629
A.12.1	HTML_BBCodeParser	629
A.12.2	HTML_Common	629
A.12.3	HTML_Crypt	630
A.12.4	HTML_CSS	630
A.12.5	HTML_Form	631
A.12.6	HTML_Menu	631
A.12.7	HTML_Page	632
A.12.8	html_parse	632
A.12.9	HTML_Progress	632
A.12.10	HTML_QuickForm	633
A.12.11	HTML_QuickForm_Controller	634
A.12.12	HTML_Select	634
A.12.13	HTML_Select_Common	635
A.12.14	HTML_Table	635
A.12.15	HTML_Table_Matrix	636
A.12.16	HTML_Table_Sortable	636
A.12.17	HTML_Template_Flexy	637
A.12.18	HTML_Template_IT	638
A.12.19	HTML_Template_PHPLIB	639
A.12.20	HTML_Template_Sigma	639
A.12.21	HTML_Template_Xipe	640
A.12.22	HTML_TreeMenu	641
A.12.23	Pager	641
A.12.24	Pager_Sliding	642
A.12.25	tidy	642
A.13	HTTP	642
A.13.1	HTTP	642
A.13.2	HTTP_Client	643
A.13.3	HTTP_Download	643
A.13.4	HTTP_Header	643
A.13.5	HTTP_Request	644
A.13.6	HTTP_Session	644
A.13.7	HTTP_Upload	644
A.13.8	HTTP_WebDAV_Client	645
A.13.9	HTTP_WebDAV_Server	645
A.14	Bilder	646
A.14.1	FreedImage	646

A.14.2	Image_Barcode	646
A.14.3	Image_Color	646
A.14.4	Image_GIS	646
A.14.5	Image_Graph	647
A.14.6	Image_GraphViz	647
A.14.7	Image_IPTC	647
A.14.8	Image_Remote	648
A.14.9	Image_Text	648
A.14.10	Image_Tools	648
A.14.11	Image_Transform	649
A.14.12	imagemagick	649
A.15	Internationalisierung	650
A.15.1	fribidi	650
A.15.2	l18N	650
A.15.3	l18Nv2	650
A.15.4	idn	651
A.15.5	Translation	651
A.15.6	Translation2	651
A.16	Logging	652
A.16.1	Log	652
A.16.2	Log_Parser	652
A.17	Mail	652
A.17.1	Mail	652
A.17.2	mailparse	653
A.17.3	Mail_IMAP	653
A.17.4	Mail_Mbox	653
A.17.5	Mail_Mime	654
A.17.6	Mail_Queue	654
A.17.7	POP3	655
A.17.8	vpopmail	655
A.18	Mathematik	655
A.18.1	Math_Basex	655
A.18.2	Math_Complex	655
A.18.3	Math_Fibonacci	656
A.18.4	Math_Histogram	656
A.18.5	Math_Integer	656
A.18.6	Math_Matrix	656
A.18.7	Math_Numerical_RootFinding	657

A.18.8	Math_Quaternion	657
A.18.9	Math_RPN	658
A.18.10	Math_Stats	658
A.18.11	Math_TrigOp	658
A.18.12	Math_Vector	658
A.19	Netzwerke	659
A.19.1	cvsclient	659
A.19.2	cyrus	659
A.19.3	kadm5	659
A.19.4	mqseries	659
A.19.5	nettools	659
A.19.6	Net_CheckIP	660
A.19.7	Net_Curl	660
A.19.8	Net_Cyrus	660
A.19.9	Net_Dict	660
A.19.10	Net_Dig	661
A.19.11	Net_DIME	661
A.19.12	Net_DNS	661
A.19.13	Net_Finger	661
A.19.14	Net_FTP	662
A.19.15	Net_GameServerQuery	662
A.19.16	Net_Geo	662
A.19.17	Net_Gopher	662
A.19.18	Net_Ident	663
A.19.19	Net_IMAP	663
A.19.20	Net_IPv4	663
A.19.21	Net_IPv6	663
A.19.22	Net_IRC	664
A.19.23	Net_LDAP	664
A.19.24	Net_LMTP	664
A.19.25	Net_NNTP	665
A.19.26	Net_Ping	665
A.19.27	Net_POP3	665
A.19.28	Net_Portscan	665
A.19.29	Net_Server	666
A.19.30	Net_Sieve	666
A.19.31	Net_SmartIRC	666
A.19.32	Net_SMTTP	668

A.19.33	Net_Socket	668
A.19.34	Net_Traceroute	669
A.19.35	Net_URL	669
A.19.36	Net_UserAgent_Detect	669
A.19.37	Net_UserAgent_Mobile	669
A.19.38	Net_Whois	670
A.19.39	opendirectory	670
A.19.40	spread	670
A.19.41	tcpwrap	671
A.19.42	uuid	671
A.19.43	Beschreibung	671
A.19.44	yaz	671
A.20	Zahlen	671
A.20.1	Numbers_Roman	671
A.20.2	Numbers_Words	672
A.21	Zahlungen	673
A.21.1	cybercash	673
A.21.2	cybermut	673
A.21.3	Payment_Clieop	673
A.21.4	Payment_DTA	673
A.21.5	Payment_Process	673
A.21.6	spplus	674
A.21.7	TCLink	674
A.22	PEAR	674
A.22.1	PEAR	674
A.22.2	PEAR_Frontend_Gtk	675
A.22.3	PEAR_Frontend_Web	675
A.22.4	PEAR_Info	675
A.22.5	PEAR_PackageFileManager	676
A.23	PHP	676
A.23.1	apd	676
A.23.2	bcompiler	677
A.23.3	ffi	677
A.23.4	Inline_C	677
A.23.5	memcache	678
A.23.6	mono	678
A.23.7	perl	678
A.23.8	PHPDoc	678

A.23.9	PHPUnit	679
A.23.10	PHPUnit2	679
A.23.11	PHP_CompatInfo	679
A.23.12	PHP_Fork	679
A.23.13	python	681
A.23.14	Validate	681
A.23.15	Var_Dump	682
A.23.16	vld	682
A.23.17	Xdebug	683
A.24	Verarbeitung	683
A.24.1	FSM	683
A.25	Wissenschaft	684
A.25.1	Science_Chemistry	684
A.26	Streams	684
A.26.1	bz2_filter	684
A.26.2	oggvorbis	684
A.26.3	openal	685
A.26.4	Stream_SHM	685
A.26.5	Stream_Var	685
A.26.6	zlib_filter	685
A.27	Strukturen	686
A.27.1	Games_Chess	686
A.27.2	OLE	686
A.27.3	Structures_DataGrid	686
A.27.4	Structures_Graph	687
A.27.5	Text_Statistics	687
A.27.6	Tree	687
A.28	System	688
A.28.1	statgrab	688
A.28.2	System_ProcWatch	688
A.28.3	System_Socket	689
A.29	Text	689
A.29.1	enchant	689
A.29.2	lzf	689
A.29.3	panda	690
A.29.4	Text_Diff	690
A.29.5	Text_Password	690

A.29.6	Text_Wiki	691
A.29.7	xdiff	691
A.30	Werkzeuge und Hilfsprogramme	691
A.30.1	crack	691
A.30.2	fann	691
A.30.3	PECL_Gen	692
A.30.4	PhpDocumentor	692
A.30.5	SPL	693
A.30.6	Valkyrie	694
A.31	Webdienste	694
A.31.1	Services_ExchangeRates	694
A.31.2	Services_Weather	694
A.31.3	SOAP	695
A.31.4	SOAP_Interop	695
A.31.5	UDDI	695
A.31.6	XML_RPC	695
A.32	XML	696
A.32.1	XML_Beautifier	696
A.32.2	XML_CSSML	696
A.32.3	XML_DTD	696
A.32.4	XML_fo2pdf	697
A.32.5	XML_FOAF	697
A.32.6	XML_HTMLSax	697
A.32.7	XML_image2svg	698
A.32.8	XML_NITF	698
A.32.9	XML_Parser	699
A.32.10	XML_RDDL	699
A.32.11	XML_RSS	699
A.32.12	XML_SaxFilters	700
A.32.13	XML_Serializer	701
A.32.14	XML_sql2xml	701
A.32.15	XML_Statistics	702
A.32.16	XML_SVG	702
A.32.17	XML_svg2image	702
A.32.18	XML_Transformer	702
A.32.19	XML_Tree	703
A.32.20	XML_Util	703
A.32.21	XML_Wddx	703

A.32.22 XML_XPath	703
A.32.23 XML_XSLT_Wrapper	704
A.32.24 XML_XUL	705
B phpDocumentor	707
B.1 Einführung	707
B.2 Kommentare zur Dokumentation	707
B.3 Übersicht über die Tags	709
B.3.1 abstract	709
B.3.2 access	710
B.3.3 author	712
B.3.4 category	712
B.3.5 copyright	712
B.3.6 deprecated	713
B.3.7 example	713
B.3.8 filesource	714
B.3.9 final	714
B.3.10 global	715
B.3.11 ignore	716
B.3.12 inheritdoc (inline)	716
B.3.13 internal, internal (inline)	717
B.3.14 licence	717
B.3.15 link	718
B.3.16 link (inline)	718
B.3.17 name	719
B.3.18 package	719
B.3.19 param	720
B.3.20 return	721
B.3.21 see	722
B.3.22 since	722
B.3.23 static	723
B.3.24 staticvar	723
B.3.25 subpackage	724
B.3.26 todo	724
B.3.27 uses	724
B.3.28 var	725
B.3.29 version	726

B.4	Tabelle der Tags	726
B.5	Das Werkzeug phpDocumentor verwenden	728
C	Schnelleinstieg in Zend Studio	737
C.1	Version 4	737
C.2	Über den Schnelleinstieg in Zend Studio	737
C.3	Über Zend	737
C.4	Übersicht über den Zend-Studio-Client	738
C.4.1	Komponenten des Studios	738
C.4.2	Client-Server-Konfiguration	739
C.4.3	Installation und Registrierung	739
C.5	Eine Datei bearbeiten	741
C.5.1	Eine Datei bearbeiten	741
C.6	Mit Projekten arbeiten	742
C.6.1	Vorteile beim Arbeiten mit Projekten	742
C.6.2	Ein Projekt erstellen	742
C.7	Den Debugger ausführen	743
C.7.1	Der interne Debugger	744
C.7.2	Der Remote-Debugger	745
C.7.3	Einen URL debuggen	745
C.8	Den Studio-Server für Remote-Debugging und -Profilng konfigurieren	746
C.9	Den Profiler ausführen	747
C.10	Produktunterstützung	749
C.10.1	Wo Sie Unterstützung erhalten	749
C.11	Die Hauptfunktionen	750
	Stichwortverzeichnis	751