

Inhalt

Vorwort	19
---------------	----

1 Einführung 21

1.1 Kleines Einstiegsprojekt zum Kennenlernen	21
1.1.1 Ein erstes XML-Dokument	21
1.1.2 Standardausgabe im Webbrower	22
1.1.3 Wohlgeformtheit ist ein Muss	23
1.1.4 Gültige Dokumente per DTD oder Schema	24
1.1.5 Formatierte Datenausgabe	26
1.2 XML – universale Metasprache und Datenaustauschformat	28
1.2.1 Unabhängigkeit von Anwendungen und Plattformen	28
1.2.2 SGML -> HTML -> XML	28
1.2.3 Lob des Einfachen	29
1.2.4 Inhaltsbeschreibungssprache	29
1.2.5 Trennung von Inhalt und Form	30
1.2.6 Vom Dokumentformat zum allgemeinen Datenformat ...	30
1.2.7 Globale Sprache für den Datenaustausch	31
1.2.8 Interoperabilität	32
1.3 Übersicht über die Sprachfamilie XML	33
1.3.1 Kernspezifikationen	33
1.3.2 Ergänzende Spezifikationen	34
1.3.3 Programmierschnittstellen	35
1.3.4 XML-Anwendungen	35
1.4 XML-Editoren und Entwicklungsumgebungen	36
1.4.1 Spezialeditoren für XML	36
1.4.2 Schema- und Stylesheet-Designer	38
1.4.3 Entwicklungsumgebungen mit XML-Unterstützung	39
1.4.4 XML-Dokumente über Standardanwendungen	40
1.4.5 Parser und andere Prozessoren	41
1.5 Anwendungsbereiche	42
1.5.1 XML-Vokabulare	42
1.5.2 Datenaustausch zwischen Anwendungen	44
1.5.3 Verteilte Anwendungen und Webdienste	45

2 XML-Bausteine und Regeln 47

2.1 Aufbau eines XML-Dokuments	47
--------------------------------------	----

2.1.1	Entitäten und Informationseinheiten	47
2.1.2	Parsed und unparsed	48
2.1.3	Die logische Sicht auf die Daten	49
2.1.4	Der Prolog	51
2.1.5	Zeichenkodierung	52
2.1.6	Standalone or not	53
2.1.7	XML-Daten: der Baum der Elemente	54
2.1.8	Start-Tags und End-Tags	54
2.1.9	Elementtypen und ihre Namen	55
2.1.10	Regeln für die Namensgebung	56
2.1.11	Elementinhalt	57
2.1.12	Korrekte Schachtelung	58
2.1.13	Attribute	59
2.2	Die Regeln der Wohlgeformtheit	60
2.3	Elemente oder Attribute?	61
2.4	Reservierte Attribute	61
2.4.1	Sprachidentifikation	62
2.4.2	Leerraumbehandlung	62
2.5	Entitäten und Verweise darauf	63
2.5.1	Eingebaute und eigene Entitäten	63
2.5.2	Zeichenentitäten	63
2.6	CDATA-Sections	64
2.7	Kommentare	65
2.8	Verarbeitungsanweisungen	65
2.9	Namensräume	66
2.9.1	Das Problem der Mehrdeutigkeit	66
2.9.2	Eindeutigkeit durch URLs	67
2.9.3	Namensraumname und Präfix	68
2.9.4	Namensraumdeklaration und QNames	68
2.9.5	Einsatz mehrerer Namensräume	69
2.9.6	XML Version 1.1	70

3 Dokumenttypen und Validierung 73

3.1	Metasprache und Markup-Vokabulare	73
3.1.1	Datenmodelle	73
3.1.2	Selbstbeschreibende Daten und Lesbarkeit	74
3.1.3	Dokumenttyp-Definition – DTD	74
3.1.4	XML Schema	75
3.1.5	Vokabulare	75
3.2	Regeln der Gültigkeit	76

3.3	DTD oder Schema?	77
3.4	Definition eines Dokumentmodells	77
3.4.1	Interne DTD	78
3.4.2	Externe DTD	79
3.5	Deklarationen für gültige Komponenten	80
3.5.1	Vokabular und Grammatik der Informationseinheiten ...	80
3.5.2	Syntax der Dokumenttyp-Deklaration	80
3.5.3	Syntax der Elementtyp-Deklaration	81
3.5.4	Beispiel einer DTD für ein Kursprogramm	82
3.5.5	Inhaltsalternativen	84
3.5.6	Uneingeschränkte Inhaltsmodelle	85
3.5.7	Gemischter Inhalt	86
3.5.8	Inhaltsmodell und Reihenfolge	86
3.5.9	Kommentare	87
3.5.10	Die Hierarchie der Elemente	87
3.6	Dokumentinstanz	88
3.7	Attributlisten-Deklaration	89
3.7.1	Aufbau einer Attributliste	89
3.7.2	Attributtypen und Vorgaberegelungen	90
3.7.3	Verwendung der Attributlisten	91
3.8	Verweis auf andere Elemente	93
3.9	Verwendung von Entitäten	93
3.9.1	Interne Entitäten	94
3.9.2	Externe Entitäten	94
3.9.3	Notationen und ungeparste Entitäten	96
3.9.4	Verwendung von Parameterentitäten	97
3.9.5	Interne Parameterentitäten	97
3.9.6	Externe Parameterentitäten	97
3.10	Formen der DTD-Deklaration	98
3.10.1	Öffentliche und private DTDs	98
3.10.2	Kombination von externen und internen DTDs	99
3.10.3	Bedingte Abschnitte in externen DTDs	100
3.11	Zwei DTDs in der Praxis	101
3.11.1	Das grafische Format SVG	101
3.11.2	SMIL	104

4 Inhaltsmodelle mit XML Schema 111

4.1	XML Schema – der XML-basierte Standard	111
4.1.1	Defizite von DTDs	111
4.1.2	Anforderungen an XML Schema	112

4.1.3	Die Spezifikation des W3C für XML Schema	112
4.2	Erster Entwurf eines Schemas	113
4.2.1	Verknüpfung von Schema und Dokument	116
4.2.2	Der Baum der Schema-Elemente	116
4.2.3	Elemente und Datentypen	117
4.2.4	Komplexe Typen mit und ohne Namen	117
4.2.5	Sequenzen	119
4.2.6	Vorgegebene und abgeleitete Datentypen	119
4.2.7	Wie viel wovon?	119
4.3	Genereller Aufbau eines XML-Schemas	120
4.3.1	Das Vokabular	120
4.3.2	Die Komponenten eines XML-Schemas	120
4.4	Datentypen	121
4.4.1	Komplexe Datentypen	122
4.4.2	Inhaltsmodelle und Partikel	122
4.4.3	Erweiterbarkeit durch Wildcards	123
4.4.4	Einfache Typen	124
4.4.5	Benannte oder anonyme Typen	125
4.4.6	Vorgegebene und benutzerdefinierte Datentypen	125
4.4.7	XML Schema – Datentypen – Kurzreferenz	126
4.4.8	Werter Raum, lexikalischer Raum und Facetten	129
4.4.9	Ableitung durch Einschränkung	130
4.4.10	Muster und reguläre Ausdrücke	131
4.4.11	Grenzwerte	132
4.4.12	Listen und Vereinigungen	133
4.4.13	Facetten der verschiedenen Datentypen	134
4.5	Definition der Struktur des Dokuments	135
4.5.1	Deklaration von Elementen	135
4.5.2	Attribute	137
4.5.3	Elementvarianten	138
4.5.4	Namensräume in XML Schema	138
4.5.5	Zielnamensraum	139
4.5.6	Umgang mit lokalen Elementen und Attributen	141
4.6	Häufigkeitsbestimmungen	144
4.7	Default-Werte für Elemente und Attribute	145
4.8	Kompositoren	146
4.8.1	<xsd:sequence>	147
4.8.2	<xsd:all>	147
4.8.3	<xsd:choice>	148
4.8.4	Verschachtelte Gruppen	148
4.9	Arbeit mit benannten Modellgruppen	148

4.10	Definition von Attributgruppen	150
4.11	Schlüsselemente und Bezüge darauf	151
4.11.1	Eindeutigkeit	151
4.11.2	Bezüge auf Schlüsselemente	152
4.12	Kommentare	154
4.13	Ableitung komplexer Datentypen	155
4.13.1	Erweiterungen komplexer Elemente	155
4.13.2	Einschränkung komplexer Elemente	156
4.13.3	Steuerung der Ableitung von Datentypen	157
4.13.4	Abstraktionen	158
4.13.5	Gemischtwaren	159
4.13.6	Leere oder Nichts	160
4.13.7	Wiederverwendbarkeit	161
4.14	Designvarianten	162
4.14.1	Babuschka-Modelle	163
4.14.2	Stufenmodelle	164
4.15	Übernahme von Schema-Definitionen	165
4.15.1	Schemas inkludieren	166
4.15.2	Schemas importieren	167
4.15.3	Zuordnung von Schemas in XML-Dokumenten	172
4.16	XML Schema – Kurzreferenz	173

5 Navigation und Verknüpfung 183

5.1	Datenauswahl mit XPath	183
5.1.1	Baummodell und XPath-Ausdrücke	184
5.1.2	Vom Dokument zum Knotenbaum	184
5.1.3	Dokumentreihenfolge	186
5.1.4	Knotentypen	187
5.1.5	Lokalisierungspfade	188
5.1.6	Ausführliche Schreibweise	190
5.1.7	Lokalisierungsstufen und Achsen	190
5.1.8	Knotentest	194
5.1.9	Filtrern mit Prädikaten	195
5.1.10	Test von XPath-Ausdrücken	195
5.1.11	XPath 1.0-Funktionen	197
.5.2	XPath 2.0	201
5.2.1	Erweitertes Datenmodell	201
5.2.2	Neue Konstrukte für Ausdrücke	202
5.2.3	Neue Datentypen	203
5.2.4	Neue Operatoren	203

5.2.5	Die erweiterte Funktionenbibliothek	204
5.3	Verknüpfungen mit XLink	214
5.3.1	Mehr als Anker in HTML	214
5.3.2	Beziehungen zwischen Ressourcen	215
5.3.3	Link-Typen und andere Attribute	216
5.3.4	Beispiel für einen einfachen Link	218
5.3.5	Beispiel für einen Link vom Typ »extended«	219
5.3.6	XLink-Anwendungen	220
5.4	XBase	220
5.5	Über XPath hinaus: XPointer	221
5.5.1	URIs und Fragmentbezeichner	222
5.5.2	XPointer-Syntax	222
5.5.3	Das Schema element0	223
5.5.4	Das Schema xmlns()	224

6 Datenausgabe mit CSS 227

6.1	Cascading Stylesheets für XML	229
6.2	Arbeitsweise eines Stylesheets	229
6.3	Anlegen von Stylesheets	231
6.4	Vererben und Überschreiben	233
6.5	Selektortypen	234
6.6	Attributselektoren	235
6.7	Kontext- und Pseudoselektoren	235
6.8	Schriftauswahl und Textformatierung	236
6.8.1	Absolute Maßeinheiten	236
6.8.2	Relative Maßeinheiten	237
6.8.3	Prozentangaben	237
6.8.4	Maßangaben über Schlüsselwörter	237
6.9	Farbauswahl	237
6.10	Blöcke, Ränder, Rahmen, Füllung und Inhalt	238
6.11	Stylesheet-Kaskaden	240
6.12	Auflösung von Regelkonflikten	240
6.13	Zuordnung zu XML-Dokumenten	241
6.14	Schwächen von CSS	242

7 Umwandlungen mit XSLT 243

7.1	Sprache für Transformationen	243
7.1.1	Bedarf für Transformationen	243
7.1.2	Grundlegende Merkmale von XSLT	245

7.1.3	XSLT-Prozessoren	246
7.1.4	Die Elemente und Attribute von XSLT	247
7.1.5	Verknüpfung zwischen Stylesheet und Dokument	249
7.1.6	Das Element <stylesheet>	250
7.1.7	Top-Level-Elemente	251
7.1.8	Template-Regeln	252
7.1.9	Attributwert-Templates	253
7.1.10	Zugriff auf die Quelldaten	255
7.2	Ablauf der Transformation	255
7.2.1	Startpunkt Wurzelknoten	256
7.2.2	Anwendung von Templates	256
7.2.3	Rückgriff auf versteckte Templates	257
7.2.4	Auflösung von Template-Konflikten	258
7.3	Stylesheet mit nur einer Template-Regel	258
7.4	Eingegebene Template-Regeln	259
7.5	Designalternativen	260
7.6	Kontrolle der Knotenverarbeitung	262
7.6.1	Benannte Templates	263
7.6.2	Template-Auswahl mit XPath-Mustern	264
7.6.3	Kontext-Templates	266
7.6.4	Template-Modi	266
7.7	Datenübernahme aus der Quelldatei	268
7.8	Nummerierungen	270
7.8.1	Einfach	270
7.8.2	Mehrstufig	270
7.8.3	Zusammengesetzt	272
7.9	Verzweigungen und Wiederholungen	273
7.9.1	Bedingte Ausführung von Templates	273
7.9.2	Wahlmöglichkeiten	274
7.9.3	Schleifen	275
7.10	Sortieren und Gruppieren von Quelldaten	278
7.10.1	Sortierschlüssel	278
7.10.2	Sortierreihenfolge	280
7.11	Parameter und Variablen	281
7.11.1	Parameterübergabe	281
7.11.2	Globale Parameter	282
7.11.3	Lokale und globale Variablen	282
7.11.4	Eindeutige Namen	283
7.11.5	Typische Anwendungen von Variablen in XSLT	284
7.11.6	Rekursive Templates	288
7.12	Hinzufügen von Elementen und Attributen	290

7.12.1	Elemente und Attribute aus vorhandenen Informationen erzeugen	291
7.12.2	Attributlisten	292
7.12.3	Texte und Leerräume	293
7.12.4	Kontrolle der Ausgabe	293
7.13	Zusätzliche XSLT-Funktionen	294
7.13.1	Zugriff auf mehrere Quelldokumente	294
7.13.2	Zahlenformatierung	296
7.13.3	Liste der zusätzlichen Funktionen in XSLT	297
7.14	Mehrfache Verwendung von Stylesheets	298
7.14.1	Stylesheets einfügen	299
7.14.2	Stylesheets importieren	299
7.15	Übersetzungen zwischen XML-Vokabularen	300
7.15.1	Diverse Schemas für gleiche Informationen	301
7.15.2	Angleichung durch Transformation	302
7.16	Umwandlung von XML in HTML und XHTML	303
7.16.1	Datenübernahme und Ergänzungen	304
7.16.2	Generieren von CSS-Stylesheets	305
7.16.3	Aufbau einer Tabelle	306
7.16.4	Transformation in XHTML	307
7.16.5	XHTML-Module	307
7.16.6	Allgemeine Merkmale von XHTML	308
7.16.7	Aufbau eines XHTML-Dokuments	308
7.16.8	Automatische Übersetzung	309
7.17	Exkurs zu XHTML und HTML 5	310
7.18	XSLT-Editoren	311
7.19	Kurzreferenz zu XSLT 1.0	313
7.20	XSLT 2.0	321
7.20.1	Die wichtigsten Neuerungen	321
7.20.2	Neue Funktionen in XSLT 2.0	327
7.20.3	Neue Elemente	328

8 Formatierung mit XSL 333

8.1	Transformation und Formatierung	333
8.2	Formatierungsobjekte	334
8.3	Baum aus Bereichen – Areas	335
8.4	XSL-Bereichsmodell	336
8.4.1	Block-Bereiche und Inline-Bereiche	336
8.4.2	XSL und CSS	337
8.5	Testumgebung für XSL	337

8.6	Aufbau eines XSL-Stylesheets	339
8.6.1	Baum der Formatierungsobjekte	339
8.6.2	Seitenaufbau	340
8.6.3	Seitenfolgen	341
8.6.4	Einfügen von Fließtext	342
8.6.5	Blockobjekte	342
8.7	Verknüpfung mit dem Dokument und Ausgabe	344
8.8	Inline-Formatierungsobjekte	346
8.9	Ausgabe von Tabellen	347
8.9.1	Tabellenstruktur	347
8.9.2	Zellinhalte	348
8.10	Listen	350
8.11	Gesucht: visuelle Editoren	352
8.12	Übersicht über die Formatierungsobjekte von XSL	352
8.12.1	Übergeordnete Objekte	352
8.12.2	Blockformatierung	354
8.12.3	Inline-Formatierung	354
8.12.4	Tabellenformatierung	355
8.12.5	Listenformatierung	356
8.12.6	Formatierung für Verknüpfungen	356
8.12.7	Out-of-Line-Formatierung	357
8.12.8	Andere Objekte	357

9 Abfragen mit XQuery 359

9.1	Datenmodell und Verfahren	359
9.1.1	Zur Syntax	361
9.1.2	Instanzen des Datenmodells	361
9.1.3	W3C-Empfehlungen zu XQuery	364
9.2	Abfragepraxis	365
9.2.1	XQuery-Modul	365
9.2.2	Zugriff über das Web	366
9.3	FLWOR-Ausdrücke	367
9.3.1	Variablen in XQuery	370
9.3.2	Steuerung der Ausgabe	371
9.4	Fortgeschrittene Optionen	372
9.4.1	Auswertung zweier verbundener Dokumente	372
9.4.2	Kollektionen auswerten	373
9.4.3	Benutzerdefinierte Funktionen	374
9.5	Implementierungen	375

10 Einstieg in XForms 379

10.1	XForms – Empfehlung und Implementierungen	380
10.2	Testübung	383
10.3	Aufbau des Datenmodells	385
10.4	Steuerelemente	389
10.4.1	Texteingabe	390
10.4.2	Das Element submit	391
10.4.3	Textbereiche	391
10.4.4	Radiobuttons	392
10.4.5	Checkboxen	393
10.4.6	Optionengruppen	394
10.4.7	Schaltflächen	395
10.4.8	Switch	396
10.4.9	Steuerung der Steuerelemente	398
10.5	Datenhandhabung	399
10.5.1	Wiederholte Elemente	399
10.5.2	Einfügen und Löschen	400
10.5.3	Übertragungswege	402
10.5.4	Mehrere Formulare in einem Dokument	403
10.5.5	Datenbindung mit bind	404
10.5.6	Werte übernehmen	404
10.6	Ereignisse und Aktionen	406
10.7	Kurzreferenz XForms 1.1	408
10.7.1	Model-Elemente	408
10.7.2	Kern-Steuerelemente	409
10.7.3	Steuerelement-Container	409
10.7.4	Elemente in Steuerelementen	409
10.7.5	Aktionen	410
10.7.6	Ereignisattribute	410
10.7.7	XForms-Ereignisse	411

11 Programmierschnittstellen für XML 413

11.1	Abstrakte Schnittstellen: DOM und SAX	413
11.2	Document Object Model (DOM)	415
11.2.1	DOM Level	416
11.2.2	Objekte, Schnittstellen, Knoten und Knotentypen	417
11.2.3	Die allgemeine Node-Schnittstelle	417
11.2.4	Knotentypen und ihre Besonderheiten	420
11.2.5	Zusätzliche Schnittstellen	420

11.2.6	Zugriff über Namen	421
11.2.7	Verwandtschaften	422
11.2.8	Das Dokument als DOM-Baum	422
11.2.9	Document – die Mutter aller Knoten	424
11.2.10	Elementknoten	425
11.2.11	Textknoten	426
11.2.12	Attributknoten sind anders	426
11.2.13	Dokumentfragmente	427
11.2.14	Fehlerbehandlung	427
11.3	DOM-Implementierungen	427
11.4	Die MSXML-Implementierung von DOM	428
11.4.1	Schnittstellen in MSXML	429
11.4.2	Erweiterungen für Laden und Speichern	431
11.4.3	Erweiterungen der Node-Schnittstelle	432
11.5	Fingerübungen mit DOM	434
11.5.1	Daten eines XML-Dokuments abfragen	435
11.5.2	Zugriff über Elementnamen	440
11.5.3	Zugriff auf Attribute	441
11.5.4	Abfrage über einen Attributwert	443
11.5.5	Fehlerbehandlung	444
11.5.6	Neue Knoten einfügen	445
11.5.7	Neue Elementknoten	448
11.5.8	Neue Attributknoten	449
11.5.9	Unterelementknoten und Textknoten	449
11.5.10	Request und Response	450
11.6	Alternative zu DOM: Simple API for XML (SAX)	451
11.6.1	Vergesslicher Beobachter am Datenstrom	452
11.6.2	SAX2 unter Java	452
11.6.3	Der Kern der SAX-Schnittstellen	454
11.6.4	ContentHandler	455
11.6.5	Attribute	456
11.6.6	SAX2-Erweiterungen	457
11.6.7	Hilfsklassen	459
11.6.8	SAXParser und XMLReader	460
11.6.9	Konfigurieren des Parsers	461
11.6.10	Kleine Lagerauswertung mit SAX	463
11.6.11	Aufruf des Parsers	466
11.6.12	Fehlerbehandlung	467
11.6.13	SAX-Beispiel 1	469
11.6.14	Beispiel 2	472
11.6.15	SAX und DOM	474

11.7	Arbeit mit XML-Klassen in VB.Net	474
11.7.1	XML-Architektur im .NET Framework	475
11.7.2	Lesen von XML-Daten	476
11.7.3	XMLReader im Vergleich zum SAX-Reader	477
11.7.4	Arbeitsweise von XmlReader	478
11.7.5	XML-Dokument mit XMLTextReader auswerten	478
11.7.6	Lesen von XML-Fragmenten	482
11.7.7	Validierung anhand von XML-Schemas oder DTDs	484
11.7.8	Schreiben von XML-Daten	486
11.7.9	XmlTextWriter	490
11.7.10	XML-Serialisierung und -Deserialisierung	494

12 Kommunikation zwischen Anwendungen 501

12.1	XML-Webdienste	502
12.1.1	Gemeinsame Nutzung von Komponenten	502
12.1.2	Offen gelegte Schnittstellen	502
12.1.3	Endpunkte	502
12.2	Beispiel für einen Webdienst	503
12.2.1	Webdienst mit ASP.NET	503
12.2.2	Einrichten eines Webdienstes	504
12.2.3	Webmethoden	507
12.2.4	Test des Webdienstes	507
12.2.5	Aufruf einer Methode	508
12.2.6	Nutzen des Webdienstes über eine Anwendung	509
12.2.7	Einfügen des Verweises auf den Webdienst	510
12.2.8	Proxyklasse	511
12.3	Nachrichten mit SOAP	512
12.3.1	Ein Rahmen für Nachrichten	512
12.3.2	Grundform einer SOAP-Nachricht	513
12.4	Dienstbeschreibung	516
12.4.1	Das WSDL-Vokabular	516
12.4.2	WSDL unter ASP.NET	517
12.5	Webdienste registrieren und finden	520
12.5.1	UDDI	520
12.5.2	Disco	521
12.5.3	Safety first!	521

13 XML in Office-Anwendungen 523

13.1	XML in Office 2010	524
13.1.1	Der Standard Office Open XML	524
13.1.2	Open XML für Excel	525
13.1.3	Open XML für Word	529
13.2	Die Alternative OpenDocument	530
13.3	Einsatz benutzerdefinierter Schemas in Office 2010	534
13.4	XML-Markup in Word 2010	535
13.4.1	Schema für eine Teilnehmerliste	537
13.4.2	Zuordnen des Schemas	538
13.4.3	Zuordnen von XML-Elementen zu Textteilen	541
13.4.4	Eingabe von Attributwerten	543
13.4.5	Speicheroptionen für XML-Daten	544
13.4.6	Öffnen mit XSLT-Stylesheets	546
13.5	XML-Markup in Excel 2010	547
13.5.1	Zuordnen eines Schemas	547
13.5.2	Optionen beim Öffnen von XML-Dokumenten	548
13.5.3	Daten als XML-Tabelle übernehmen	548
13.5.4	XML-Tabellenbereiche	551
13.5.5	XML-Zuordnungen	552
13.5.6	Datenaktualisierung	552
13.5.7	Öffnen als schreibgeschützte Arbeitsmappe	553
13.5.8	Verwenden von XSLT-Stylesheets	554
13.5.9	Datenquelle und Tabelle manuell verknüpfen	556
13.5.10	XmlMap-Objekte	560
13.5.11	Tabelle auf Basis eines eigenen Schemas	560
13.5.12	Fehlererkennung	561
13.5.13	XML-Dokumente erzeugen	562
13.5.14	Schema-Einschränkungen	562
13.6	XML-basierte Formulare mit InfoPath 2010	562
13.6.1	Werkzeug für dynamische Formulare	563
13.6.2	Fingerübung mit InfoPath	564
13.6.3	Formular mit eigener Datenstruktur	564
13.6.4	Formularentwurf »from Scratch«	568
13.6.5	XPath-Ausdrücke für Berechnungen	569
13.6.6	Schema-Limits	570
13.6.7	Validierung per Schema	570
13.6.8	Zusatzprüfungen	570
13.6.9	Formularsichten	571
13.6.10	Veröffentlichung von Formularen	571

Inhalt

13.6.11 Vorlagenarchiv	571
13.6.12 Formulare ausfüllen	574
13.6.13 Speichern der eingegebenen Daten	575
13.6.14 Austausch mit anderen Anwendungen	576
14 Mapping – von XML oder nach XML	579
14.1 Codegenerierung für Transformationen	579
14.1.1 Oberfläche und Dateiformate	580
14.1.2 Funktionsbibliotheken	582
14.1.3 Von Schema zu Schema	583
14.2 Datenausgabe	585
14.3 Stylesheet-Generierung	586
14.4 Eigene Funktionen	587
14.5 Mapping von Datenbankdaten	588
14.6 Mapping für Excel-Tabellen	590
14.7 EDIFACT und ANSI X12	591
Anhang	593
A Webressourcen	593
A.1 Webseiten für Entwickler	593
A.2 Liste von Empfehlungen des W3C	595
A.3 Liste von wichtigen Namensräumen des W3C	597
B Glossar	599
Index	611