

# 1 Inhalt

<b>1</b>	<b>Einstieg in Flex 4.0</b>	<b>1</b>
1.1	Next Generation Internet	1
1.1.1	Die Flash Platform	2
1.1.2	Die Protagonisten	2
1.2	Die Bestandteile von Flex 4	3
1.3	Entwicklung und Aufbau einer Flex-Anwendung	5
1.4	Flex für den Desktop: Adobe AIR	6
<b>2</b>	<b>Die Entwicklungsumgebung: Flash Builder 4</b>	<b>7</b>
2.1	Die verschiedenen Projektarten	11
2.2	Ein neues Projekt anlegen und ausführen	11
<b>3</b>	<b>MXML</b>	<b>13</b>
3.1	Struktur und Syntax	13
3.1.1	MXML-Grundgerüst und Zeichenkodierung	14
3.1.2	Kommentare in MXML	15
3.1.3	Platzieren von Komponenten	17
3.1.4	MXML-Dateinamen	17
3.1.5	Aufbau von MXML-Dokumenten	18
3.2	MXML und ActionScript	20
<b>4</b>	<b>ActionScript</b>	<b>23</b>
4.1	Einführung	23
4.1.1	Über ActionScript	23

4.1.2 ActionScript im Umfeld von Flex	25
4.1.3 Spuren legen mit trace()	26
4.1.4 Kommentare in ActionScript	27
4.2 Einsatz von ActionScript	28
4.2.1 Inline: Innerhalb von MXML-Tag-Attributen	29
4.2.2 Nested: verschachtelt in MXML-Tags	30
4.2.3 Block: MXML Script Blocks	30
4.2.4 Ausgelagert: ActionScript in separater Datei	32
4.2.5 Klassen: Externe Definitionsdateien	34
4.3 ActionScript 3.0 im Vergleich	35
4.3.1 Was gibt es Neues in ActionScript 3.0	35
4.4 Sprachelemente und Syntax	39
4.4.1 Variablen und Datentypen	39
4.4.2 Klassen, Eigenschaften und Methoden	43
4.5 Interfaces programmieren	45
4.6 Vergleich: ActionScript, PHP und Java	49
<b>5 Das Event-Modell von Flex</b>	<b>51</b>
5.1 Einführung	51
5.1.1 Ereignisfluss: das Event-Modell verstehen	52
5.2 Event-Handler	53
5.2.1 Events in MXML abfangen	53
5.2.2 Events in ActionScript abfangen	56
5.2.3 Welche Events sind verfügbar?	58
5.2.4 Die Event-Klasse	59
5.3 Event Propagation	61
5.3.1 Unterschied zwischen target und currentTarget	62
5.3.2 Die addEventListener()-Methode	64
5.3.3 Priorität von Events	65
5.3.4 Propagation in Event-Kette stoppen	66
5.4 Event-Unterklassen	67
5.4.1 MouseEvent	69
5.4.2 KeyboardEvent	71
5.4.3 Gesten und ihre Events	73

---

5.5	Events für Fortgeschrittene	74
5.5.1	Eigene Event-Klasse erstellen	74
5.5.2	Events manuell feuern	75
<b>6</b>	<b>Die Flex-Bausteine: Komponenten</b>	<b>79</b>
6.1	Komponenten	79
6.1.1	Controls	80
6.1.2	Layout-Container	94
6.1.3	Navigators	95
6.2	Textkomponenten	99
6.2.1	Flash Text Engine (ab Flash Player 10)	99
6.2.2	Text Layout Framework (TLF)	100
6.2.3	Die Text-Komponenten	100
6.2.4	Das Text-Objektmodell	101
6.2.5	Einfachen Text in MXML	102
6.2.6	Eingabe von Text (TextInput und TextArea)	104
6.2.7	Texteingabe und die relevanten Events	105
6.2.8	Textformatierung in ActionScript (TextFlow)	107
6.2.9	Templates: TextFlow und Datenbindung	108
6.2.10	Mehrspaltiger Text	109
6.2.11	Formatierten Text importieren (importFromXML und importFromString)	110
6.2.12	TextFlow exportieren	111
6.3	List-basierte Komponenten	112
6.3.1	List	113
6.3.2	Tree (Halo)	118
6.3.3	DataGrid	121
6.3.4	AdvancedDataGrid	123
6.3.5	ItemRenderer	123
6.3.6	ItemEditor	127
6.3.7	DropDownList und ComboBox	131
6.3.8	DataGroup und SkinnableDataContainer	133
6.4	Layout	134
6.4.1	Die möglichen Layouts	136
6.4.2	Absolute Positionierung (BasicLayout)	137
6.4.3	Constraint-based Layout (BasicLayout)	138

6.4.4 TileBasedLayout	139
6.4.5 Verschachteln von Containern	140
6.4.6 Rahmen und Abstand	142
6.4.7 Scrollbalken	143
<b>7 Medienintegration</b>	<b>147</b>
7.1 Einführung	147
7.2 Grafiken und Bilder laden	149
7.2.1 SWF-Animationen und Skripte laden	154
7.2.2 Ladevorgang überwachen	158
7.3 Einbinden von Sound	159
7.4 Einbinden von Video	161
7.5 Restriktionen beim Laden externer Medien	163
<b>8 Anpassen der Benutzeroberfläche</b>	<b>165</b>
8.1 Einleitung	165
8.2 Styles	166
8.2.1 Instance/ Inline Style	167
8.2.2 CSS: Class Styles und Type Styles	168
8.2.3 CSS-Syntax	169
8.2.4 CSS in Flex und die allgemeine CSS-Spezifikation	172
8.2.5 Globale Styles	173
8.2.6 Einsatz von Styles in MXML und externen CSS-Dateien	174
8.2.7 Einsatz von Styles in ActionScript	175
8.2.8 Verfügbare Style-Eigenschaften	178
8.2.9 Formate von Style-Eigenschaften	179
8.2.10 Vererbung und Dominanz	181
8.3 Component Skinning	184
8.3.1 Beispiel	185
8.4 Skinning mit Flash Catalyst	192
8.4.1 Die Designvorgabe und das Artwork (Photoshop)	193
8.4.2 Die Komponente in Flash Catalyst zum Leben erwecken	193
8.4.3 Import in Flash Builder	194
8.5 Fonts	196
8.5.1 Fonts nutzen und einbetten	197
8.5.2 Probleme beim Nutzen und Einbetten von Fonts	200

---

8.5.3 Problembehandlung bei Halo-Komponenten	202
8.5.4 Die Bereiche der Zeichensätze anpassen	203
8.6 Runtime CSS	205
<b>9 States, Effects und Transitions</b>	<b>209</b>
9.1 States	209
9.1.1 Mit States arbeiten (MXML)	211
9.1.2 Mögliche Modifikationen innerhalb eines States	214
9.1.3 States ableiten	222
9.1.4 States in Komponenten und Modulen	223
9.1.5 States und ihre Events	224
9.1.6 States verfeinern	224
9.2 Effects	224
9.2.1 Effects erstellen und anwenden	225
9.2.2 Die Trickkiste: Übersicht der verfügbaren Effects und Trigger	229
9.2.3 Effects kombinieren	230
9.2.4 Effects und ihre Events	232
9.2.5 Sonderfälle	234
9.3 Transitions	241
9.3.1 Transitions mit States verknüpfen	241
9.3.2 Event-Kette von States und Transitions	247
9.3.3 Eins nach dem anderen: Action Effects	249
9.3.4 Fine-Tuning: Filter	251
<b>10 Weitere Werkzeuge des Flex-Frameworks</b>	<b>255</b>
10.1 PopUp	255
10.2 Drag and Drop	257
10.2.1 Events einer Drag-and-Drop-Operation	258
10.2.2 Custom Drag and Drop (für nicht List-basierte Komponenten)	261
10.3 Drucken in Flex	264
10.3.1 Einfaches Drucken	264
10.3.2 Drucken eines DataGrids mit Druckvorlage	266
<b>11 Umgang mit lokalen Daten</b>	<b>269</b>
11.1 Einführung	269
11.1.1 Daten aktuell halten: Datenbindung (Data Binding)	269
11.1.2 Daten richtig strukturieren und bereitstellen: Data Model	271

11.1.3 Daten aufbereiten und formatieren: Data Formatter	273
11.1.4 Daten auf Gültigkeit überprüfen: Data Validation	274
11.1.5 Datenfluss	276
11.2 Data Binding	277
11.2.1 Uni- und bidirektionale Inline-Datenbindung	278
11.2.2 Die Tag-Variante für die Datenbindung	281
11.2.3 Datenbindungen mit ActionScript realisieren	283
11.2.4 Eigene Objekteigenschaften für Datenbindungen aktivieren	288
11.2.5 ActionScript innerhalb einer Datenbindung	295
11.3 Data Model	296
11.3.1 Model, View und Controller (MVC)	297
11.3.2 Data Model in Flex	299
11.4 Data Formatting	314
11.4.1 Formatter in MXML anlegen	315
11.4.2 Formatter in ActionScript realisieren	317
11.4.3 Fehler in Formatter abfangen	318
11.4.4 Übersicht der Formatter-Klassen von Flex	320
11.4.5 Custom Formatter programmieren	323
11.5 Data Validation	325
11.5.1 Validator in MXML anlegen	327
11.5.2 Validator in ActionScript realisieren	328
11.5.3 Übersicht der Validator-Klassen von Flex	329
11.5.4 Pflichtfelder	342
11.5.5 Manuelles Auslösen eines Validators	343
11.5.6 Fehlermeldungen anpassen	347
11.5.7 Custom Validator programmieren	347
11.6 LocalConnection	350
<b>12 Externe Datenquellen und Data Centric Development</b>	<b>359</b>
12.1 Einleitung	359
12.2 HTTP-basierte Kommunikation	361
12.2.1 HTTPService	362
12.2.2 WebService	370
12.2.3 RemoteObject	379
12.3 Data Centric Development	380
12.3.1 Codegenerierung	382

---

12.3.2 CallResponder	383
12.3.3 Fiber Model	383
12.4 PHP und ZendAMF	386
12.4.1 Voraussetzungen	386
12.4.2 Ausgangspunkt PHP	387
12.4.3 Ausgangspunkt Flex	397
12.5 Ein unschlagbares Team: Flex, Java, BlazeDS und Hibernate	417
12.5.1 Die Serverseite	419
12.5.2 Flex-Client: Manager	436
12.5.3 Flex-Client: Frontend	443
12.5.4 Zusammenfassung	448
12.6 Messaging mit BlazeDS	448
12.6.1 Chat-Anwendung mit mehreren Themen	449
<b>13 Komponenten und Module entwickeln</b>	<b>457</b>
13.1 Einleitung	457
13.1.1 Vorteile der modularen Entwicklung	457
13.1.2 Komponenten	459
13.1.3 Module	460
13.1.4 Komponenten vs. Module	461
13.1.5 Workflow und Aufbau einer modularen Flex-Anwendung	461
13.2 Eigene Komponenten entwickeln	462
13.2.1 Composite Components (MXML)	463
13.2.2 Extended Components (ActionScript)	473
13.2.3 Template Components (MXML)	486
13.2.4 Komponentenentwicklung für Fortgeschrittene	491
13.2.5 Component Lifecycle – Darstellung und Aktualisierungsprozess von Komponenten	495
13.2.6 Eine Custom Component entwickeln	529
13.2.7 Komponenten kompilieren und verteilen	543
13.3 Module	551
13.3.1 ModuleLoader	552
13.3.2 ModuleManager	555
13.3.3 Kommunikation mit Modulen	556
13.3.4 Application Domain vs. Module Domain	562

<b>14 Gute Programmierung, schlechte Programmierung</b>	<b>565</b>
14.1 So nicht: eng gekoppelte Programmierung	566
14.2 Lose gekoppelte Programmierung	570
<b>15 Flex für den Desktop (Adobe AIR)</b>	<b>577</b>
15.1 Einführung in Adobe AIR	577
15.2 Entwicklung einer AIR-Anwendung	578
15.2.1 AIR-Projekt im Flash Builder erstellen	578
15.2.2 Die AIR-Komponenten	579
15.2.3 Spezielle AIR-Funktionen und nicht sichtbare Komponenten	584
15.2.4 Veröffentlichung von AIR-Anwendungen	590
15.2.5 Exportieren von AIR-Anwendungen	590
15.3 Beispielanwendungen	591