
Inhaltsverzeichnis

1	Vorwort	1
1.1	Für wen dieses Buch gedacht ist.	1
1.2	Einleitung	1
1.3	Danksagung	2
2	Einführung in Web Components	3
2.1	Von Dokumenten zu Anwendungen: neue Ansätze für neue Anforderungen im Web	3
2.2	Web-Component-Standards im Überblick	5
2.2.1	Template-Element	5
2.2.2	Shadow DOM	5
2.2.3	Custom-Elements	6
2.2.4	HTML-Imports	6
2.3	Templating	6
2.3.1	Konventionelle Templating-Systeme	6
2.3.2	Das Template-Element	7
2.3.3	Performance von Templates	8
2.4	CSS für Komponenten	9
2.4.1	Isolierte Styles dank Shadow DOM	11
2.4.2	Eingefügte Inhalte, Shadow und CSS	13
2.5	Eigene Tags für Komponenten	16
2.5.1	Bisherige Ansätze	17
2.5.2	Custom Elements	18
2.5.3	Das <content>-Element	19
2.5.4	Auf eingebettete Inhalte zugreifen mit getDistributedNodes	21
2.5.5	Auf Elemente innerhalb von <content> zugreifen	24
2.5.6	Eigene Attribute und Methoden	25
2.5.7	Existierende HTML-Elemente erweitern	28
2.6	Komponenten wiederverwenden	29
2.6.1	Anwendungsdokument und Komponentendokument	29
2.7	Ein Grundgerüst für eigene Komponenten	31

3	Entwickeln mit Web Components	33
3.1	»Hallo Web Components!« – Ein erstes Beispiel	33
3.2	Der Anwendungsentwurf	34
3.3	Die Anwendungslogik	35
3.3.1	Aufbau und Gliederung der Benutzeroberfläche	35
3.3.2	Aufgabenliste	35
3.3.3	Notizen	35
3.3.4	Implementierung	36
3.4	Komponenten für unsere Anwendung	36
3.4.1	Testen unserer Komponenten	37
3.4.2	Das Testsystem	37
3.5	Das todo-item	39
3.5.1	Tests	39
3.5.2	Implementierung	42
3.6	Link-List	44
3.7	Tab-Komponente	48
3.8	Fazit	52
4	Polymer-Grundlagen	53
4.1	Aufbau einer Polymer-Anwendung	53
4.1.1	Das Polymer-Framework	53
4.1.2	Die Polymer-Polyfills: webcomponents.js	54
4.1.3	Eigene Elemente, fremde Elemente und Pages	54
4.2	Polymer-Elemente registrieren	55
4.2.1	Hallo-Polymer	55
4.2.2	Lifecycle-Callbacks	56
4.3	Local DOM und Light DOM	57
4.4	Properties als Element-API	58
4.4.1	Attribute, Properties und Eigenschaften	60
4.4.2	Namensgebung	60
4.4.3	Datentypen	60
4.5	Data-Binding	61
4.5.1	Was ist Data-Binding?	61
4.5.2	Bidirektionale Bindings	63
4.6	Events in Polymer-Elementen	65
4.6.1	Event-Behandlung in Polymer	65
4.6.2	Dynamische Listener	67
4.6.3	Eigene Events definieren	69

4.6.4	Gesten in Polymer	70
4.6.5	Die track-Geste	71
4.6.6	Events und das Shadow DOM	73
4.6.7	Event-Retargeting und Shady DOM	76
4.7	Styling und Theming in Polymer-Elementen	76
4.7.1	Besonderheiten von CSS für Shady DOM	76
4.7.2	CSS-Custom-Properties und Mixins	79
4.7.3	Styles dynamisch ändern und abrufen	82
4.7.4	Einschränkungen des Polyfills	83
4.7.5	Styles und Custom-CSS-Properties auf Dokumentenebene	85
4.7.6	Styles wiederverwenden mit Style Modules	86
4.7.7	Zusätzliche Bibliotheken, Polymer-Elemente und CSS	88
5	Fortgeschrittene Techniken in Polymer	91
5.1	Besonderheiten der Polyfills	91
5.1.1	Eigenschaften des Polyfills-Custom-Elements	91
5.1.2	Eigenschaften des Polyfills-HTML-Imports	92
5.1.3	Eigenschaften des Polyfills Shadow DOM (»Shady DOM«)	92
5.1.4	Shady DOM als Polyfill-Alternative	93
5.2	Globale Einstellungen	95
5.2.1	Shadow oder Shady DOM wählen	96
5.2.2	Verzögerte Registrierung mit lazyRegister	96
5.3	Das Local DOM und die Polymer-DOM-API	97
5.3.1	Light DOM und Distributed Children	97
5.3.2	Effektive Kindelemente	99
5.3.3	Observer im Local DOM	101
5.4	Properties im Detail	104
5.4.1	Die Datentypen Boolean, Object und Array	104
5.4.2	Properties und Konstruktoren	105
5.4.3	Properties vom Typ Object und Standardwerte	105
5.4.4	Properties und Attribute synchronisieren	107
5.4.5	Properties vor Änderung schützen	108
5.4.6	Abgeleitete Properties	109
5.4.7	Property-Änderungen behandeln mit Observern	110
5.4.8	Zusammengesetzte Observer	111
5.4.9	Komplexe Observer und komplexe zusammengesetzte Properties	112
5.5	Data-Binding	116
5.5.1	Property- und Attribut-Bindings im Detail	116
5.5.2	HTML in Text-Bindings	118
5.5.3	Uni- und bidirektionale Bindings	121

5.5.4	Fortgeschrittenes Data-Binding	123
5.5.5	Computed Bindings	124
5.5.6	Data-Binding für native Elemente	128
5.5.7	Data-Binding-Helfer	129
5.6	Elemente erweitern mit Behaviors.	151
5.6.1	Zweck und Umfang von Behaviors	151
5.6.2	Ein Beispiel: JsonApiClientBehavior	151
5.6.3	Reihenfolge von Behaviors und Callbacks	155
5.6.4	Behaviors erweitern.	156
5.6.5	Namespacing.	157
5.6.6	Behaviors und einmalige Aktionen	157
6	Testen von Polymer-Komponenten	159
6.1	Die Grenzen von QUnit	159
6.2	Der Web Components Tester	161
6.3	WCT mit einem beliebigen Webserver	161
6.4	WCT als Kommandozeilen-Werkzeug	163
6.5	Tests mit WCT	164
6.5.1	Assertions und Expectations	164
6.5.2	Fixtures	165
6.5.3	Asynchrone Tests	167
6.5.4	Stubs	167
6.5.5	Integrationstests	173
6.6	Die iron-test-helpers	177
6.6.1	MockInteractions	177
6.6.2	Weitere Test-Helfer.	180
6.7	Mehrere Test-Suites	181
6.8	Testen gegen Shady DOM und Shadow DOM	182
6.9	Barrierefreiheit testen mit der a11y-suite	182
7	Eine Polymer-Anwendung von A bis Z	185
7.1	Anwendungsentwurf	185
7.1.1	Aufbau der Anwendung	185
7.1.2	Suchfunktion.	186
7.1.3	Filmdetails.	187
7.1.4	Favoriten.	187
7.1.5	Merkliste.	187
7.1.6	Archiv	187
7.2	Projektstruktur	188
7.3	Die Abhängigkeiten vorbereiten	189

7.4	Das Anwendungsdokument.	189
7.4.1	Das Grundgerüst	189
7.4.2	Struktur und Layout.	190
7.4.3	Das vorläufige Anwendungsdokument.	192
7.5	Routing und Pages.	193
7.5.1	Routing-Grundlagen: Fragment-URLs und History API.	193
7.5.2	Routing in Polymer mit dem App-Router.	198
7.6	Die Elemente für unsere Anwendung.	202
7.6.1	Die Datenschicht	202
7.6.2	Die Ergebniskachel.	210
7.6.3	Die sortierbare Tabelle.	221
7.6.4	Die sortierbare Liste.	228
7.7	Die Pages für unsere Anwendung	230
7.7.1	Suche	230
7.7.2	Filmdetails	234
7.7.3	Favoriten	236
7.7.4	Merkliste	236
7.7.5	Archiv	237
7.8	Das vollständige Anwendungsdokument.	238
7.9	Die Anwendung für den Produktiveinsatz vorbereiten.	239
A	Anhang – Exkurs: Bower	243
A.1	Was ist Bower?	243
A.2	Warum Bower?	243
A.3	Bower installieren	244
A.4	Bower nutzen	245
A.5	Abhängigkeiten versionieren	249
A.6	Eigene Bower-Module bereitstellen	250
	Schlusswort	253
	Index	255