

E-Books selbst gebaut

Jens Pönisch

27. Oktober 2010



1 Überblick und Technik

1.1 Was sind E-Books?

.....

- Digitalisierung von Bibliotheksbeständen und Archiven?
- Bücher als Datei, die man am Computer liest?
- Hörbücher???
- Bücher, die man auf speziellen Lesegeräten (E-Book-Reader) liest?

Uns interessieren hier Bücher für E-Book-Reader.

Schreibweisen: *E-Book* (Duden), *eBook*

1.2 Geschichte

.....

- Seit 1970-er Jahren Digitalisierung von Büchern (1971: Projekt Gutenberg)
- 1990 Sony Data Discman (Mini-CDs, LCD), geringes Buchangebot, schlechtes Display
- 1991 erste elektronische Bücher auf Diskette für Apple-Laptops: Jurassic Park, Alice in Wonderland, The Hitchhikers Guide through the Galaxy
- Mehrere weitere Anläufe in den 90er Jahren ohne großen Erfolg
- 2004 Sony: erster E-Book-Reader mit elektronischer Tinte
- Durchbruch 2007 mit dem Kindle (Amazon)
- Seitdem viele Geräte

1.3 Warum spezielle Geräte

Netbooks, PDAs, Mobiltelefone mit Display sind doch verfügbar?

- Leicht und handlich
- Buchgefühl :-)
- Geringer Stromverbrauch (8.000 - 15.000 Seitenwechsel)
- Displayauflösung
- Kontrast
- Bequeme Handhabung
- Ideal für Reisen

1.4 Eigenschaften gängiger eBook-Reader

- Displaygröße: 5-6 Zoll (japanische Taschenbücher)
- Auflösung 600x800 Pixel (164-200 dpi - Taschenbuchqualität)
- E-Ink-Technologie - Stromverbrauch nur beim Blättern
- Betriebssystem meist embedded Linux
- Verschiedene Schriftgrößen
- Nur Schwarz-weiß-Darstellung
- 4 bis 16 Graustufen
- Interner Speicher 128 MB bis 4 GB
- Gewicht ca. 150-250g
- Zusatzfeatures wie Notizen, Sound, Touchpanel, Speicherkarten, Abonnementfunktion für Zeitschriften, Online-Buchkauf
- Übertragung der Bücher per USB oder Internetzugang (WLAN, Mobilfunk)

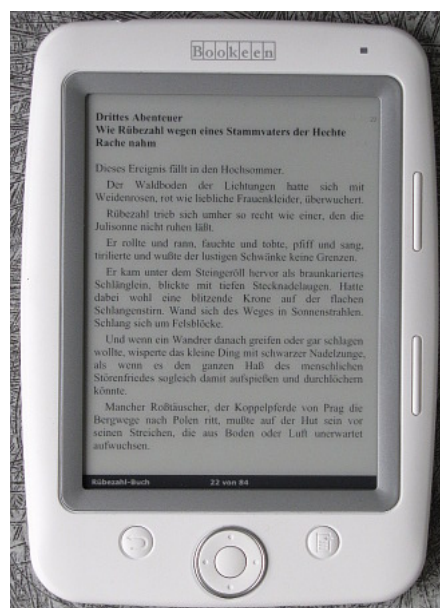


Abbildung 1.1: Cybook Opus - Buchtext

Spezialgeräte mit größeren Displays (768x1024 Pixel und größer) vereinzelt verfügbar.

Reader-Vergleich: http://wiki.mobileread.com/wiki/E-book_Reader_Matrix

1.5 Beispiel: Cybook Opus

Display:

- 600x800 Pixel
- 4 Graustufen
- 9 Schriftgrößen
- 7,5x10 cm

Größe:

- 11x15 cm
- 150g mit Batterie

Prozessor, Speicher und OS:

- ARM 400 MHz
- 1 GB Flash intern und steckbare Micro-SD-Card
- Linux, Sourcecode kann von <http://www.bookeen.com/> heruntergeladen werden.

Formate:

- EPUB (alternativ: MobiPocket lt. Herstellerseite?)
- Adobe Digital Editions DRM
- HTML
- PDF
- Text
- Bilder in gängigen WWW-Formaten

Preis: ca. 200 € (Mai 2010)



Abbildung 1.2: Cybook Opus - Buchwahl

1.6 Beispiel: eFox-Reader

Display:

- TFT, Auflösung unbekannt
- Farbdisplay
- 4 Schriftgrößen
- 9x15 cm

Sonstige Eigenschaften:

- 13x20 cm
- Netzteil, Kopfhörer mitgeliefert
- 270g mit Batterie
- 3 GB Flash intern und steckbare Micro-SD-Card
- USB-Anschluss (Storage Device)
- Laufzeit: ca. 5 Stunden, Ladezeit: ca. 2 Stunden

Formate:

- EPUB (kein Inhaltsverzeichnis, CSS, PRE, Tabellen!) - vermutlich *FBReader*
- kein DRM
- HTML
- PDF (kein Vergrößern möglich)
- Text
- Bilder in gängigen WWW-Formaten mit Diashow, Vergrößerung
- Videos im Flash- und DivX-Format
- MP3-Audiodateien

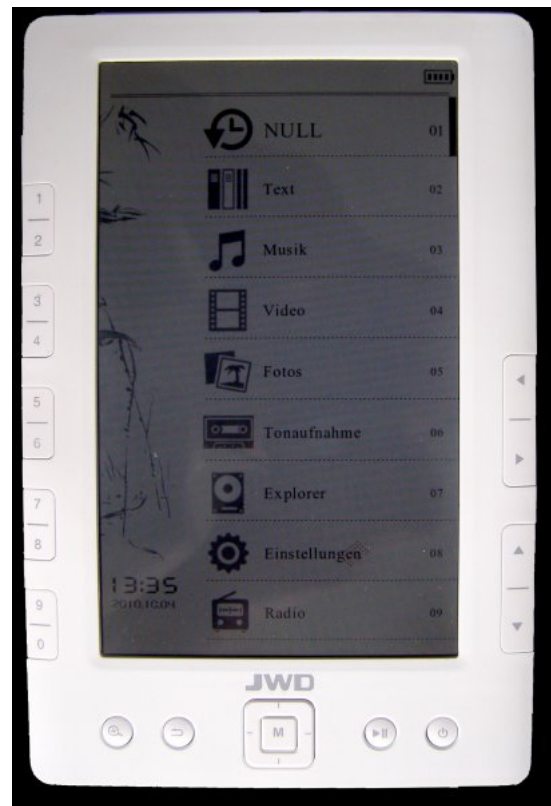


Abbildung 1.3: eFox-Reader - Hauptmenü

MJPEG-Videos von Digitalkameras können wie folgt umgewandelt werden:

```
1 ffmpeg -i mpeg.avi -vcodec libxvid -aspect 4:3 -r 15 -ar 48000 xvid.avi
```

Anmerkung 1.1. *Fazit: Als E-Book-Reader nur sehr bedingt geeignet, eher zum Anzeigen von Fotos und Videos.*

Preis: ca. 60 € (Oktober 2010)

Achtung: Laut Internetforen hat die Firma praktisch keinerlei Support, Impressum fehlt ebenfalls.

1.7 Nachteile von E-Book-Readern

- Kleines Display
- Nur schwarz-weiße Darstellung
- relativ langsam
- schlechtere Typographie gegenüber gedruckten Büchern
- kein Scrollen, nur Blättern

- kein Multimedia (Filme) - Audio aber teilweise möglich
- Hyperlinks nur innerhalb eines Dokuments
- keine Multifunktionsgeräte
- Preis relativ hoch
- Etwa 6-12% geringere Lesegeschwindigkeit gegenüber klassischen Büchern: Studie (<http://www.useit.com/alertbox/ipad-kindle-reading.html>)

1.8 Auswahlkriterien beim Kauf

- Buchformat (Mobipocket XOR EPUB)
- Displaygröße
- Einstellbare Schriftgrößen (3 sind zuwenig!)
- Gewicht (max. 200 Gramm - iPad ist zu schwer als E-Book)
- Stromverbrauch
- Zusatzfunktionen wie Notizen, Lesezeichen, Audio
- Datenübertragungsmöglichkeiten

Preise (Frühjahr 2010):

- Displaygröße 5 Zoll: ab 200 € (Weltbild: 99 €)
- Displaygröße 6 Zoll: 250-350 € (Thalia: 139 €, ab 28. Oktober 2010)
- Große Displays: 400-800 € (Pocketbook: 329 € ab November 2010)
nur wenige Geräte verfügbar/angekündigt, Rückgang zugunsten von Tablet-PCs

Im Herbst 2010 weitere neue Geräte angekündigt: Preise 140 bis 160 €.

1.9 E-Paper

auch: *ePaper*, *eInk*

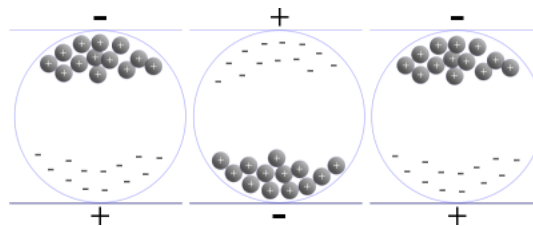


Abbildung 1.4: E-Paper - Funktionsweise

- Derzeit wichtigste Technologie: *Elektrophorese*
- Leitend beschichtete Glasplatten mit 10-100 μm Abstand
- Dazwischen dunkles Öl und helle, elektrisch geladene Partikel (Titanoxid)
- Durch entsprechend gepolte Spannung bewegen sich Partikel zur anziehenden Platte.
- Partikel oben: weiß (Reflexion), unten schwarz (Absorption durch Öl)
- Wirkung wie Kondensator
- Durch geringe Dicke der Glasplatten sehr guter Kontrast und Ablesewinkel

- Energieverbrauch nur beim Umschalten der Ladung
- Derzeit wichtigster Hersteller: E Ink Corporation

Entwicklungen: farbige Displays, schnellere Umschaltzeiten, flexible Displays.

2 E-Book-Formate und -Anzeigeprogramme

2.1 E-Book-Formate

.....

Im Wesentlichen zwei konkurrierende Formate:

- Mobipocket (Kindle, Amazon)
- EPUB (alle anderen, Software: Adobe Digital Edition)

Im russischen Raum noch ein weiteres Format:


- FictionBook (XML-Dialekt für Inhalt und Packaging-Format)

Zusätzlich meist HTML, Text, PDF und Bilder möglich.

Im Weiteren wird nur das EPUB-Format genauer betrachtet.

2.2 Anzeige von E-Books auf dem Computer

.....

- Sichten, Recherche
- Verwalten von Bibliotheken
- Backup des E-Book-Readers
- Kein E-Book-Reader vorhanden
- Drucken
- Konvertieren in andere Formate (EPUB  Mobipocket)

2.3 Adobe Digital Editions (ADE)

.....

- Referenzimplementierung
- Anzeige und Verwaltung von E-Books
- Drucken
- DRM-Unterstützung
- Werkzeug für Download und Übertragung gekaufter Bücher
- DRM-Aktivierung von E-Book-Readern
- Eingesetzte Software auf E-Book-Reader
- Nur für Windows und MacOSX, derzeit nicht für Linux (obwohl fast alle E-Book-Reader Linux einsetzen!)

2.4 Firefox-Plug-in

.....

<https://addons.mozilla.org/en-US/firefox/addon/45281/>

- Einfach zu installieren
- Anzeige und Bibliotheksverwaltung
- Recht tolerant bei Fehlern
- Kein DRM

2.5 Calibre

.....

<http://calibre-ebook.com/>

- Linux (nicht auf Lenny), MacOSX, Windows
- Gute Anzeigefunktionalität
- Konvertierung zwischen verschiedenen Formaten
- Erstellen von E-Books aus HTML, DOC und Text
- Erstellen von E-Books aus Newstickern (Heise, New York Times, ...)
- Übertragung auf E-Book-Reader
- Keine DRM-Unterstützung

Umfangreichste freie Reader-Software.

2.6 FBReader

.....

<http://www.fbreader.org/>

- Linux, Windows, BSD, mobile Plattformen
- keine Tabellen, schlechte CSS-Unterstützung
- keine Fonteinbettung
- recht unausgereift, nur bedingt brauchbar

3 Quellen für E-Books

3.1 Kommerzielle Bücher







.....

- Schutz durch DRM, meist mit Adobe-Werkzeugen
- Bei Kauf erhält man personalisierte URL
- Mit Hilfe von ADE Buch in persönlicher Verschlüsselung laden
- Auf aktivierten E-Book-Reader übertragen

- E-Book ist nun personengebunden

3.2 Vor- und Nachteile von E-Books

.....

-  - leicht, kaum Platzbedarf
-  - elektronisch übertragbar, per UMTS immer verfügbar
-  - Ausleihe, Weitergabe bei DRM-Büchern kaum möglich
-  - keine Anmerkungen bei DRM-Büchern möglich
-  - bei Verlust des DRM-Schlüssels Komplettverlust
-  - viel zu teuer (10 % Preisnachlass in Deutschland) aufgrund der Einschränkungen

US-Studie zeigt:

- Akzeptierter Preis E-Book USA: 5-10 \$
- Akzeptierter Preis E-Book Deutschland: 5-9 €

Quelle (<http://www.lesen.net/ereader/studie-reader-tablets-begehrt-aber-zu-teuer-3101/>)

Aber: Nachteile betreffen nur DRM-geschützte Bücher.

3.3 Freie Bücher

.....

- Urheberrecht gilt für Literatur, Wissenschaft, Kunst.
- «Persönliche geistige Schöpfung» - Schöpfungshöhe
- Gibt dem Autor das Recht, selbst über Verwertungsrechte zu entscheiden sowie Veränderungen am Werk zu verbieten.
- Urheberrecht läuft 70 Jahre nach Tod des Urhebers bzw. nach Veröffentlichung (bei anonymen Urheber) ab.
- Anschließend werden Werke gemeinfrei.
- Anbieter kann Weitergabe der Inhalte dann nicht verbieten.
- Amtliche Werke sind stets gemeinfrei.
- Achtung: Auch Übersetzungen haben eine Schöpfungshöhe und sind damit nicht automatisch mit dem Ablauf der originalen Urheberrechte gemeinfrei!

Damit besteht die legale Möglichkeit, Inhalte zu digitalisieren und in dieser Form bereitzustellen. Der Inhalt ist dann stets gemeinfrei!

3.4 Projekt Gutenberg

.....

- <http://www.gutenberg.org>
- Digitalisierung gemeinfreier Werke und öffentliche Bereitstellung
- Derzeit ca. 30.000 Bücher
- International, aber Großteil englischsprachig
- Bereitstellung in vielen Formaten:
 - Plain text

- HTML
- EPUB
- MobiPocket
- ...

3.5 Feedbooks

.....

- <http://www.feedbooks.com>
- Große Zahl freier Bücher
- Großteil englisch
- Formate:
 - EPUB
 - MobiPocket
 - PDF
- Schönes Layout

3.6 Projekt Gutenberg-DE

.....

- <http://gutenberg.spiegel.de/>
- Projekt des Spiegel-Verlags
- Im wesentlichen deutsche Literatur, große Auswahl
- HTML (nicht wellformed!), nur zum Online-Lesen gedacht
- Rechte unklar (Verwertungsrechte für Layout werden beansprucht - fragwürdig)
- Inhalte sind gemeinfrei

Nutzung auf E-Book-Readern: Download der Kapitel und Transformation erforderlich!

3.7 Zeno.org

.....

- <http://www.zeno.org>
- Neben Büchern auch große Sammlung gemeinfreier Bilder und Postkarten

Nutzung auf E-Book-Readern: Download der Kapitel und Transformation erforderlich!

3.8 Transformation von Gutenberg-DE und Zeno.org nach EPUB

.....

Vorgehen:

1. Download der einzelnen Kapitel (Massendownload per `robots.txt` gesperrt)
2. Extrahieren der Metadaten und des eigentlichen Inhalts aus den HTML-Dateien
3. Evt. Nachladen von Abbildungen
4. Umwandeln in XHTML, Umbau der Bildverweise, Fußnoten, ...

5. Erzeugen von Strukturinformation, Inhaltsverzeichnis, Metadaten
6. Packen zur EPUB-Datei

Automatisieren ab Schritt 2: <http://xml2lms.in-chemnitz.de/epub/>

- Erzeugen einer Dateiliste
- Aufruf des Scripts:

```
1 pfad/zeno-epub.py [-f] name fileliste
2 # bzw.
3 pfad/gutenberg-de-epub.py [-f] name fileliste
```

-f bindet FreeSerif-Font ein (erweiterter Zeichensatz)

4 Das EPUB-Format

4.1 Warum Selbstbau lernen?

.....

- Gewünschte Dokumente liegen nicht im korrekten Format vor: z.B. nur als Sammlung von HTML-Seiten, als E-Learning-Paket, ...
- Layout gefällt nicht und soll umgebaut werden (nur bei unverschlüsselten E-Books möglich)
- E-Book ist «defekt»: kann mit anderen Readern nicht angezeigt und soll deshalb repariert werden (z.B. mitgeliefertes Handbuch 🙄)

4.2 Warum eigene Dateiformate?

.....

- Paketformat - ein Buch = eine Datei
- Anpassung der Schriftgröße wünschenswert -> kein starres Seitenlayout möglich
- Seitenweises Lesen
- DRM

4.3 Benötigte Werkzeuge

.....

- UTF-8-fähiger Editor (*vim*, *Eclipse*, ...)
- XML-/XHTML-Validator (*xmllint*, *Eclipse*, ...)
- XML-Transformationstools (*xsltproc*, XML-Bibliothek der Lieblingsprogrammiersprache, ...)
- Packer *zip*
- Validator *epubcheck* (und damit Java)
- E-Book-Viewer (Firefox-Plug-in, *Calibre*, ...)

4.4 Der EPUB-Standard

.....

Vom IDPF 2007 entwickelt, XML-basiert.

Fasst drei offene Standards zusammen:

Open Publication Structure (OPS) Formatierung des Inhalts

Beschreibt, welche Dokumenttypen, Multimediadaten in einem E-Book enthalten sein dürfen.

Open Packaging Format (OPF) Strukturbeschreibung der EPUB-Paketdatei

Metadaten, Liste aller Dateien und ihrer MIME-Typen, Reihenfolge

OEBPS Container Format (OCF) Paketformat

Dateistruktur und Verpackung als ZIP-File

Dazu kommen weitere Standards wie Dublin Core, XHTML, DTBook.

4.5 Notwendige Schritte

.....

1. Erstellen der Verzeichnisstruktur
2. Aufbereiten und Bereitstellen des Inhalts
3. Erstellen (Kopieren) der Dateien `mimetype` und `container.xml`
4. Erstellen der Paketbeschreibungsdatei
5. Erstellen der Inhaltsverzeichnisdatei
6. Packen des Inhalts
7. Prüfen
8. Übertragen auf den E-Book-Reader
9. Lesegenuss 😊

4.6 Verzeichnisstruktur

.....

```
1  epub -+-  mimetype
2  |
3  +- META-INF --- container.xml
4  |
5  +- data -+- content.opf
6  |
7  +- toc.ncx
8  |
9  +- Datenfiles/-verzeichnisse
```

Hervorhebung: Dateiname/-erweiterung vorgegeben.

4.7 Aufbereitung des Inhalts

.....

Der Inhalt kann in zwei Formaten bereitgestellt werden:

- XHTML 1.1
- DTBook Version 2005-2

Beachte:

- Alle Dateien im Datenverzeichnis oder Unterverzeichnissen ablegen
- Codierung: UTF-8
- Gestaltung: CSS 1.x
- Dateigröße: 300 kB unkomprimiert (100 kB komprimiert)
- Nur relative Links verwenden

4.8 Exkurs: DTBook

.....

Ebenfalls als Inhaltsformat zulässig.

Spezifikation und DTD: <http://www.daisy.org/specifications>

- Genauere semantische Beschreibung von Buchinhalten
- An HTML angelehnt, aber weitere Strukturelemente
- Beschreibt auch Seitenaufteilung
- Beschreibung von Hörbüchern möglich

Probleme:

- Buch kann nur als Einheit beschrieben und damit der Inhalt nicht auf mehrere Dateien verteilt werden.
- Zusatzinformation wie Seitenaufteilung kann kaum ausgenutzt werden.

Eher als Austauschformat für Buchproduktionen geeignet.

4.9 Datei mimetype

.....

- Enthält den Content-Type:

1

<code>application/epub+zip</code>

- Steht im Paket an erster Stelle und muss unkomprimiert gespeichert werden

Hinweis 4.1. *Die Datei ist immer identisch und kann einfach aus einer Vorlage kopiert werden.*

4.10 Verzeichnis META-INF

.....

- Muss die Datei `container.xml` enthalten.
- Bei DRM-geschützten Büchern zusätzlich `encryption.xml` und `rights.xml`
- Definiert das Datenverzeichnis und die Paketbeschreibungsdatei (Erweiterung `.opf`)
- Die Spezifikation empfiehlt den Namen OEPBS für das Datenverzeichnis, ist aber nicht zwingend.

`samples/container.xml`

```

1 <?xml version='1.0' encoding='UTF-8'?>
2 <container version="1.0" xmlns="urn:oasis:names:tc:opendocument:xmlns:container">
3   <rootfiles>
4     <rootfile media-type="application/oebps-package+xml" full-path="data/content.opf"/>
5   </rootfiles>
6 </container>

```

Hinweis 4.2. Bei immer gleicher Wahl des Datenverzeichnisses und Namens der Containerdatei ist diese Datei immer identisch und kann einfach aus einer Vorlage kopiert werden.

4.11 Paketbeschreibungsdatei - Übersicht

.....

- Liegt im Datenverzeichnis, Name in META-INF/container.xml festgelegt
- Dateierweiterung .opf
- XML-Namespace <http://www.idpf.org/2007/opf>
- Zusätzlich DC-Namespace benötigt: <http://purl.org/dc/elements/1.1/>
- Benötigt einen eindeutigen Identifier (z.B. ISBN) und eine eindeutigen URI - Syntaxregel für XML-IDs und URIs beachten!
- Beschreibt:
 - Metadaten des Buches im DC-Format
 - Enthaltene Ressourcen
 - Seitenreihenfolge
 - evt. semantische Beschreibung der Ressourcen

4.12 Paketbeschreibungsdatei - Grundstruktur

.....

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <package xmlns="http://www.idpf.org/2007/opf"
3         xmlns:dc="http://purl.org/dc/elements/1.1/"
4         version="2.0"
5         unique-identifier="bookId">
6   <metadata>
7     ...
8   </metadata>
9
10  <manifest>
11    ...
12  </manifest>
13
14  <spine>
15    ...
16  </spine>
17  ...
18 </package>

```

Anmerkung 4.3. Der Identifier *bookId* wird im Abschnitt *manifest* nochmal benötigt und sollte für jedes E-Book eindeutig sein.

4.13 Paketbeschreibungsdatei - Metadaten

```
1 <metadata>
2   <dc:identifizier id="bookId">Book-URI</dc:identifizier>
3   <dc:title>Buchtitel</dc:title>
4   <dc:language>sprachcode</dc:language>
5   ...
6 </metadata>
```

- Diese drei Felder sind Pflichtangaben.
- Für die Sprache ist der Code nach RFC3066 zu nutzen, in der einfachsten Form funktionieren also die Internet-Länderkürzel.
- Weitere mögliche Felder: `dc:creator`, `dc:publisher`, `dc:subject`, `dc:rights`, ...

Anmerkung 4.4. Die URI *Book-URI* sollte für das Buch eindeutig sein, sie wird im Inhaltsverzeichnis nochmals benötigt.

4.14 Paketbeschreibungsdatei - Manifest

```
1 <manifest>
2   <item href="inhaltsverzeichnis.ncx" id="tocId" media-type="application/x-dtbncx+xml"/>
3   <item href="datei" id="dateiId" media-type="mime-type"/>
4   ...
5 </manifest>
```

- Alle Links sind relativ bzgl. der Paketbeschreibungsdatei.
- Die Ressourcen sollten im gleichen Verzeichnis bzw. in Unterverzeichnissen bzgl. der Paketbeschreibungsdatei liegen.
- Es sind alle enthaltenen Dateien einschließlich CSS, Fonts und Bilder aufzulisten.
- Das erste Item ist das Inhaltsverzeichnis, die Dateiendung muss `.ncx` sein.
- Die Reihenfolge der Ressourcen ist beliebig.
- Es ist stets der korrekte MIME-Type anzugeben.
- Als Ressourcen sind nur Dateien in den Formaten XHTML-1.1, CSS, Opentype sowie einige Bild- und XML-Formate zulässig (s. nächster Abschnitt).

Anmerkung 4.5. Die IDs der *item*-Elemente werden im Abschnitt *spine* wieder benötigt.

4.15 Paketbeschreibungsdatei - Seitenfolge

```
1 <spine id="tocId">
2   <itemref idref="dateiId"/>
3   ...
4 </spine>
```

- Die Reihenfolge entspricht der Seitenfolge in der Anzeige.
- Alle Dateien, auf die Links zeigen, müssen auch in der Spine-Reihenfolge auftauchen.

4.16 Inhaltsverzeichnis - Aufbau

.....

Dateiendung: `.ncx`, muss als erste Datei im Manifest der Paketbeschreibung stehen.

```

1 <?xml version='1.0' encoding='UTF-8'?>
2 <ncx xmlns="http://www.daisy.org/z3986/2005/ncx/" version="2005-1">
3   <head>
4     <meta name="dtb:uid" content="Book-URI"/>
5     <meta name="dtb:depth" content="tiefe"/>
6     <meta name="dtb:totalPageCount" content="0"/>
7     <meta name="dtb:maxPageNumber" content="0"/>
8   </head>
9   <docTitle>
10    <text>Buchtitel</text>
11  </docTitle>
12  <navMap>
13    ...
14  </navMap>
15 </ncx>

```

- Die Book-URI muss mit der Paketbeschreibungsdatei übereinstimmen.
- Der Parameter `dtb:depth` gibt die anzuzeigende Verschachtelungstiefe des Inhaltsverzeichnisses an.
- Die Parameter `dtb:totalPageCount` und `dtb:maxPageNumber` werden nicht benötigt und sind auf 0 zu setzen.

4.17 Inhaltsverzeichnis - Navigation

.....

```

1 <navMap>
2   <navPoint id="navpoint-1" playOrder="1">
3     <navLabel>
4       <text>Beschriftung</text>
5     </navLabel>
6     <content src="url"/>
7   </navPoint>
8   ...
9 </navMap>

```

- Parameter `depth`: Verschachtelungstiefe
- Die IDs sind unwichtig
- Der `playorder`-Wert gibt die Anzeigereihenfolge im Inhaltsverzeichnis vor (fortlaufend ab 1 nummeriert).
- Das `navPoint`-Element darf wiederum `navPoint`-Elemente enthalten.
- Die URIs verweisen auf vorhandene Ressourcen, Fragmentangaben sind erlaubt.

4.18 Zusammenpacken der Dateien

-
- Dateiendung der Paketdatei: `.epub`
 - Datei `mimetype` zuerst unkomprimiert einpacken:

```
1 zip -X0 buch.epub mimetype
```

- Rest komprimiert dazupacken:

```
1 zip -Xr9 buch.epub META-INF data
```

4.19 Validierung

.....

Warnung 4.6. EPUB-Bücher sollten vor der Übertragung auf den Reader unbedingt validiert werden.

Defekte Bücher können zum Absturz des Reader führen!

Werkzeug: *epubcheck* (<http://code.google.com/p/epubcheck/>)

```
1 java -jar epubcheck.jar E-Book-Datei
```

Gepüft werden:

- Korrektes Dateiformat,
- Vollständigkeit der Daten,
- Korrektheit der Beschreibung aller Daten und ihrer MIME-Types,
- Korrektheit der Syntax Inhalts.

5 Gestaltung des Inhalts

5.1 XHTML

.....

- Valides XHTML 1.1 schreiben
- Anzeigebreite relativ gering
- Beachten bei Tabellen, Text neben Bildern, Listings (`pre-Element`), ...
- Einzeldokumente nicht zu lang/komplex -> lange Ladezeiten

5.2 CSS

.....

EPUB unterstützt CSS im Umfang zwischen Version 1 und 2 (-> Spezifikation).

- Externe CSS-Datei benutzen (relativer Link)
- Auf Farben verzichten (4-16 Graustufen)
- Keine dynamischen Elemente verwenden
- Maße in `em` bzw. `ex` relativ zur Schriftgröße angeben.
- Abstände im Blocksatz formatieren, Abstände verringern:

```

1  p {
2      text-align: justify;
3      margin: 0.2em 0em 0.3em 0em;
4      padding: 0mm;
5  }
6  /* Folgeabsätze einziehen */
7  p + p {
8      margin: 0.2em 0em 0.3em 0em;
9      padding: 0mm;
10     text-indent: 1em;
11  }

```

- keine absolute Positionierung verwenden.
- Attribute wie `margin-top` wollten teilweise nicht - `margin` und alle 4 Werte verwenden.

5.3 Randeinstellungen

.....

Anmerkung 5.1. *ADE produziert automatisch Seitennummern, die am rechten Rand erscheinen und bei fehlender Randeinstellung den Text überdecken.*

Funktionierende Lösung:

```

1  @page {
2      margin: 1ex 0em 1ex 0em;
3  }
4  body {
5      margin: 0ex 1em 0ex 1em;
6  }

```

Adobe empfiehlt zusätzlich eine Seitenbeschreibungdatei im XSL-FO-Format - hat sich bisher als unnötig erwiesen.

5.4 Links und Fußnoten

.....

Anmerkung 5.2. *Durch das fließende Layout und damit keiner festen Seiteneinteilung können keine Fußnoten im klassischen Sinn verwendet werden.*

Alternativen:

- Verzicht auf Fußnoten 😊
- Verwenden von Endnoten (z.B. am Ende des Kapitels) und Verlinken mit Hilfe von Ankern.

Verwendung von Links: müssen in OPF-Datei in `manifest` und auch in `spine` auftauchen - Anhang erstellen für Zusatzmaterial.

Die Verwendung von Anker und Fragment-URIs ist möglich.

5.5 Bilder

- Unterstützte Bildformate: JPEG, PNG, GIF, SVG
- SVG kann eingebettet oder als Bild verlinkt werden.
- Displaybreite von 600 Pixel beachten
- Kontrast und ausreichend scharfe Konturen

5.6 Verwenden eigener Fonts

Auf dem Reader sind Grundfonts für serifenbetonte, serifenlose und Monospaced-Schriften in verschiedenen Gewichten und Stilen verfügbar.

Wann eigene Fonts?

- Unicode-Zeichen benötigt (osteuropäische Sprachen, ...)
- Monospaced-Schrift zu breit
- Sonderschriften/Symbole benötigt
- ...

Vorgehen:

- Fonts in das OpenType-Format umwandeln (z.B. mit *FontForge*)
- Dateien in das Datenverzeichnis kopieren
- Fontressourcen im Manifest der Paketdatei erfassen (MIME-Type: `font/opentype`)
- Definition in CSS-Datei:

```
1 body {
2   font-family: Times, serif;
3   margin: 0ex 1em 0ex 1em;
4 }
5 @font-face {
6   font-family : Times;
7   font-weight : normal;
8   font-style: normal;
9   src : url(times.otf);
10 }
11 @font-face {
12   font-family : Times;
13   font-weight : normal;
14   font-style: italic;
15   src : url(times-i.otf);
16 }
17 ...
```

Links:

- <http://blog.threepress.org/2009/09/16/how-to-embed-fonts-in-epub-files/>
- <http://www.mobileread.mobi/forums/showthread.php?t=61587>

5.7 Mathematische Formeln

.....

MathML wird nicht unterstützt!

Ausweg:

- Setzen von Formeln in Mini-Tex(-Lyx)-Dokumenten
- Umwandeln in Bilder (Schriftgröße, Auflösung beachten)
- Einbinden als Bild

Alternative für parallele Anzeige mit MathML-fähigen Viewer (ungetestet):

- MathML-Formel als XML-Insel einbinden
- Ersatzdarstellung als Bild in der Paketdatei definieren

6 Alternative Formate

6.1 Alternative Formate

.....

Je nach Gerät können Dokumente in alternativen Formaten angezeigt werden:

- Textdateien
- HTML-Dateien
- PDF-Dateien
- Word-Dateien
- Bilder

Anmerkung 6.1. *Je nach Format und Gerät sind unterschiedliche Einstellungen möglich. Die weiteren Anmerkungen gelten für das Cybook Opus.*

6.2 Textdateien

.....

- Datei-Erweiterung: `.txt`
- Codierung: ISO-8859-1
- Einstellungen: Schriftart, -größe

6.3 HTML-Dateien

.....

- Datei-Erweiterung: .html
- Codierung: nur ISO-8859-1
- Einbindung von Bildern funktioniert nicht

6.4 PDF

.....

- Verschiedene Vergrößerungen möglich
- Blättern innerhalb einer Seite, aber kein Scrollen möglich!
- Normale PDF-Dateien für Formate A4/A5 praktisch unlesbar!

Aber speziell angepasste PDF-Dateien funktionieren gut:

- Seitengröße: 85 x 120 mm²
- Schriftgröße 8-9 pt
- Ränder links/rechts 1 em, oben 1 ex, unten 3 ex (Statuszeile)
- Kopf- und Fußzeilen evt. weglassen (mehr Textfläche, Seitennummer in Statuszeile)
- Enge Schrift mit kräftigen Linien wie Times verwenden, CM-Schriften sind zu schwach
- PDFTitle und PDFAuthor setzen für Inhaltsverzeichnis

LaTeX-Gerüst:

```

1 \documentclass[9pt,smallheadings]{scrartcl}
2 \setlength{\paperwidth}{85mm}
3 \setlength{\paperheight}{120mm}
4 \usepackage{times} % Enge Schrift zu empfehlen
5 \usepackage[T1]{fontenc}
6 % Ränder
7 \usepackage[top=1ex,left=1em,bottom=3ex,right=1em]{geometry}
8 % PDF-Metadaten
9 \usepackage{hyperref}
10 \hypersetup{pdfauthor={Autor},pdftitle={Titel}}
11 % Weitere Pakete wie Inputencoding, Sprache, Bilder
12 ...
13 % Seitennummern ausblenden
14 \pagestyle{empty}
15 \begin{document}
16 ...
17 \end{document}

```

7 E-Books und E-Learning

7.1 Begriff und Umsetzungen

.....

Definition und Inhalte nach Wikipedia:

- Alle Formen des Lernens, bei denen digitale Medien zum Einsatz kommen
- Verteilung des Materials

- Präsentation des Materials
- Kommunikation zwischen Lehrer und Schüler
- Kommunikation zwischen den Schülern

Schwerpunkte:

- Multimedia
- Interaktivität
- Multicodalität: Codierung des Inhalts abhängig vom Medium (Querverweise online bzw. im Buch, Animationen, ...)

Umsetzungen:

- CBT - autonome Lernprogramme
- Autorensysteme: Multimedia-Editoren, Aufzeichnungssoftware, ...
- Simulationen
- Videokonferenz, Teleteaching, Streaming
- LMS - Verwaltung von Kursen, Inhalten, Kommunikation, Abrechnung, Leistungsnachweis, ...

7.2 Anforderungen an E-Learning-Inhalte

.....

Konzentration auf elektronisches Lehrmaterial.

Anforderung	Notebook	E-Book-Reader
Lesen von strukturiertem Text	☉ - Reflexion, Ablesewinkel, Auflösung	⊕
Quelltexte, Listings	⊕ - Sofortiger Test möglich	☉ - passende Aufbereitung erforderlich
Abbildungen	⊕ - verschiedene Auflösungen möglich	☉ - passende Aufbereitung erforderlich
Formeln	☉	☉ - passende Aufbereitung erforderlich
Multimedia, Animationen	⊕	⊖ - teilweise Sound möglich
Notizen zum Inhalt	☉ - abhängig von Inhaltsformat und Software	☉ - bei entsprechendem Gerät (Tastatur/Schrifterkennung)
Bereitstellen/Aktualisieren des Inhalts	⊕ - Internet, Mobilfunk, ...	☉ - USB, Mobilfunk je nach Gerät
Kommunikation	⊕ - Internet-Telefonie, Chat, ...	⊖
Gewicht, Autonomie	☉ 2-8 Stunden Batterielaufzeit, Gewicht erträglich	⊕ etliche Tage Batterielaufzeit, Gewicht optimal

Anmerkung 7.1. *Fazit: E-Book-Reader sind für textlastige Lerninhalte (Geisteswissenschaften, Mathematik, theoretische Informatik) durchaus geeignet, aber nicht für Inhalte, die Animationen und Interaktion erfordern.*

Eine interessante Alternative könnten hier Tablet-PCs werden, die eine Lücke zwischen E-Book-Reader und Notebooks füllen.

7.3 IMS Content Packages

- Container für Lerninhalte
- Paketbeschreibung (`imsmanifest.xml`) und Inhaltsdateien
- Sequenz, Inhaltsverzeichnis im IMS-Manifest beschrieben
- Paket ist strukturierte ZIP-Datei
- keine ereignisgesteuerte Ablaufsteuerung wie SCORM

7.4 Paketbeschreibung `imsmanifest.xml`

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <manifest identifier="id"
3   schemaLocation="http://www.imsglobal.org/xsd/imscp_v1p1
4     imscp_v1p1.xsd http://www.imsglobal.org/xsd/imsmd_v1p2r
5     imsmd_v1p2p2.xsd">
6   <metadata>
7     <!-- Beschreibung nach Dublin Core -->
8     ...
9   </metadata>
10  <!-- Sequenz und Inhaltsverzeichnis -->
11  <organizations>
12    <organization identifier="org0" structure="hierarchical">
13      <title>Pakettitel</title>
14      <item identifier="org1" identifierref="re-1" isvisible="true">
15        <title>Titel Abschnitt 1</title>
16      </item>
17      ...
18    </organization>
19  </organizations>
20  <!-- Beschreibung aller enthaltener Ressourcen-->
21  <resources>
22    <resource identifier="re-1" type="webcontent" href="re-1.html">
23      <file href="re-1.html"/>
24    </resource>
25    ...
26  </resources>
27 </manifest>
```

7.5 Erzeugen von Content Packages und E-Books

- Aus gemeinsamer XML-Quelle kein Problem
- Struktur sehr ähnlich
- Frontend der Transformation identisch, Backend muss verschiedene Paketbeschreibungen erzeugen
- Behandlung dynamischer Elemente (JavaScript, Animationen, Multimedia) abfangen
- Externe Links abfangen

Realisierung z.B. durch `xml2lms.in-chemnitz.de`.

7.6 Umwandlung zwischen EPUB und Content Package

Von EPUB nach CP:

1. Auspacken
2. Erzeuge aus `content.opf` und `toc.ncx` das Manifest `imsmanifest.xml`
3. Verpacke neu

Gegenrichtung problematischer, aber realisierbar:

- Metadaten fehlen oft im `imsmanifest.xml`
- Codierung der HTML-Dateien muss auf UTF-8 geändert werden
- Transformation (X)HTML -> XHTML 1.1
- Abfangen externer Links
- Abfangen von Multimediadaten
- Inhaltsbeschreibung oft unvollständig (`imsmanifest.xml` muss nur Dateien des Inhaltsverzeichnisses aufführen)
- Aufnahme lokaler verlinkter Ressourcen in den Lesefluss (Anhang bauen)

8 Literatur

Überblick und Hardware

- [1] Pawlak: *Die Wissenschaft bei Douglas Adams*. Wiley 2010.
- [2] http://de.wikipedia.org/wiki/Elektronisches_Papier - Funktionsweise elektronisches Papier
- [3] http://wiki.mobileread.com/wiki/E-book_Reader_Matrix - Geräteübersicht
- [4] <http://www.ebookreader-info.de/uebersicht/> - Übersicht in Deutschland erhältlicher Geräte

EPUB-Spezifikation und -Anleitungen

- [5] <http://de.wikipedia.org/wiki/EPUB> - Überblick zur Spezifikation
- [6] <http://www.openebook.org/specs.htm> - OEPBS-Spezifikationen (OCF, OPF, OPS)
- [7] Dietrich: *Der kleine E-Autor*. c't 25/2009
- [8] http://www.hxa.name/articles/content/epub-guide_hxa7241_2007.html - sehr schöne Anleitung
- [9] http://www.adobe.com/devnet/digitalpublishing/epubs/EPUBBestPractices-1_0_3.epub - EPUB Best Practices von Adobe Inc.
- [10] <http://code.google.com/p/epubcheck/> - epubcheck

E-Book-Reader-Software

- [11] <http://www.epubread.com/de/> - Firefox Plug-in
- [12] <http://calibre-ebook.com> - Reader und Konvertierungsprogramm

- [13] <http://www.fbreader.org> - Reader und Konvertierungsprogramm
- [14] <http://bookworm.oreilly.com/> - Online Reader (Book upload)
- [15] <http://code.google.com/p/emerson-reader/> - Epub und Daisy Reader

Quellen für E-Books

- [16] <http://www.gutenberg.org/> - Project Gutenberg
- [17] <http://www.feedbooks.com/> - Feedbooks
- [18] <http://gutenberg.spiegel.de/> - Projekt Gutenberg-DE (Spiegel)
- [19] <http://www.zeno.org> - zeno.org

Abkürzungen

ADE *Adobe Digital Edition*

CBT *Computer Based Training*

DC *Dublin Core*

DRM *Digital Rights Management*

DTBook *Digital Talking Book*

DTD *Document Type Definition*

IDPF *International Digital Publishing Forum*

IMS *Instructional Management System*

E-Learning-Projekt, aus dem das IMS Global Learning Consortium hervor ging.

LMS *Learning Management System*

OEBPS *Open eBook Publication Structure*

OCF *OEBPS Container Format*

OPF *Open Packaging Format*

OPS *Open Publication Structure*

URI *Uniform Resource Identifier*

XML *Extensible Markup Language*

XSL-FO *XML Formatting Objects*