



## Dita4Publishers - Plugin von Eliot Kimber

<b>Über den Autor.....</b>	<b>2</b>
Impressum.....	2
<b>Installation und Konfiguration.....</b>	<b>3</b>
Installation.....	3
Einbinden der DTDs in XMLSpy.....	5
Einbinden der Dita4Publishers-DTDs in XMLMind.....	5
<b>ePub-Dateien erstellen.....</b>	<b>8</b>
ePub ohne CSS generieren.....	8
ePub über Java generieren.....	9
ePub mit Build-Datei generieren.....	10
Autor unbekannt in ePub-Vorschau.....	12
Related Links abstellen.....	12
Navigation eindeutschen.....	13
Covergraphik in Standard-Ditamap.....	13
<b>D4P-Vokabular.....</b>	<b>15</b>
D4P-Formatierungen in Concepts nutzen.....	15

## Über den Autor

---

### Kurzbiographie



Der Autor wurde 1966 in Berlin geboren. Nach der Schule 1983 erlernte er den Beruf des Elektromonteurs. Ab 1986 holte er das Abitur auf einer Berliner Volkshochschule nach. Da die DDR-Hochschullandschaft keine Magisterstudiengänge kannte, startete er 1990 sein Lehramtsstudium für Deutsch und Englisch.

Andreas Petersell ist verheiratet und hat 2 Kinder. Er arbeitet heute in Berlin als Technischer Redakteur in einer Softwarefirma.

### Kontakt

Am besten in Kontakt kommen läßt es sich über Google+ - Das Profil des Autors finden Sie unter <http://gplus.to/petersell>

## Impressum

---

Autor:	Andreas Petersell petersell@ditalog.com www.ditalog.com
Ditalog-Ausgabe:	Dita4Publishers - Plugin von Eliot Kimber
Redaktionsschluß:	13.12.2011

# Installation und Konfiguration

## Dita4Publishers installieren

Anleitung: So installieren Sie das Plugin Dita4Publishers - Version 0.9.16.

1. Downloaden Sie das Plugin unter <http://sourceforge.net/projects/dita4publishers/files/>. Die Dateisyntax ist `dita4publishers-datum.zip`.
2. Entpacken Sie die Zip-Datei. Als Ergebnis erhalten Sie folgende Zip-Dateien und Ordner:
  - html
  - documentation
  - sample\_data
  - xslt
  - dita4publishers\_toolkit\_plugins.zip
  - verschiedene d4p-users-guide-Dateien
3. Entpacken Sie die Datei `dita4publishers_toolkit_plugins.zip` in den Plugin-Ordner Ihrer Toolkit-Installation.

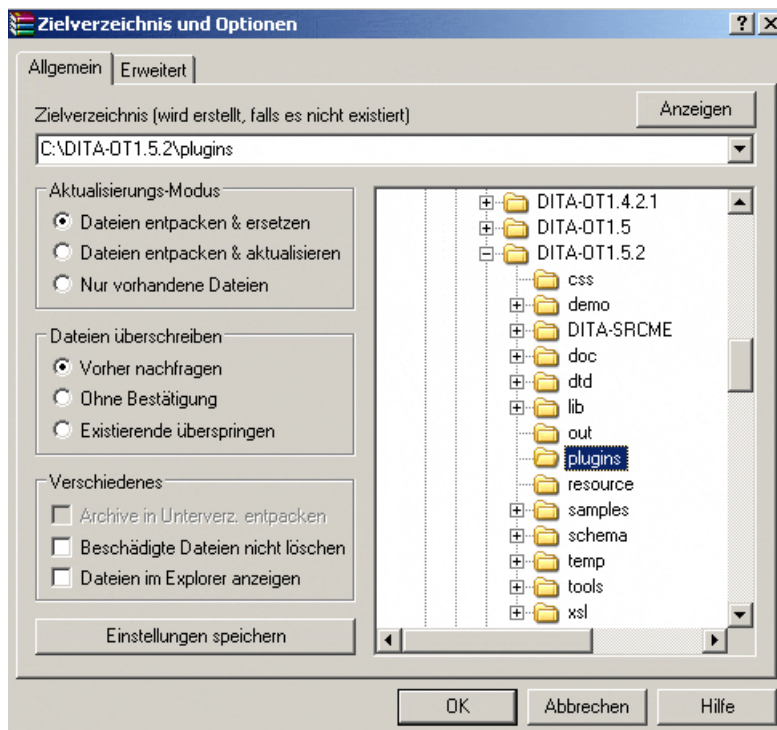


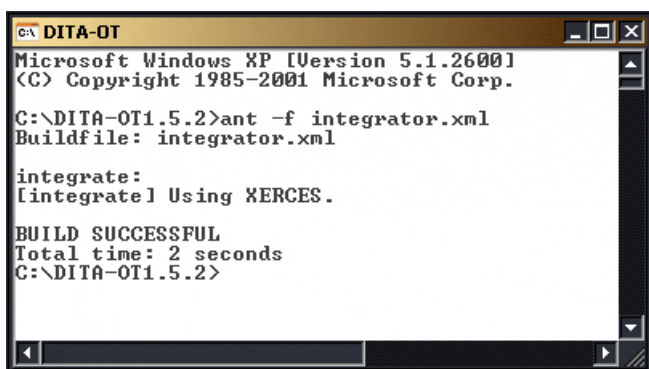
Abb. 1: plugins als Zielverzeichnis zum Entpacken

Im Ergebnis entstehen im Verzeichnis `\plugins` folgende Unterordner:

- net.sourceforge.dita4publishers.common.mapdriven
- net.sourceforge.dita4publishers.common.xslt
- net.sourceforge.dita4publishers.doctypes
- net.sourceforge.dita4publishers.enumeration-d.fo

- net.sourceforge.dita4publishers.enumeration-d.html
- net.sourceforge.dita4publishers.epub
- net.sourceforge.dita4publishers.formatting-d.fo
- net.sourceforge.dita4publishers.formatting-d.html
- net.sourceforge.dita4publishers.html2
- net.sourceforge.dita4publishers.kindle
- net.sourceforge.dita4publishers.pubContent-d.fo
- net.sourceforge.dita4publishers.pubmap.html
- net.sourceforge.dita4publishers.ruby.fo
- net.sourceforge.dita4publishers.ruby.html
- net.sourceforge.dita4publishers.sample.epub\_override
- net.sourceforge.dita4publishers.shakespeare.doctypes
- net.sourceforge.dita4publishers.shakespeare.html
- net.sourceforge.dita4publishers.variables-d.html
- net.sourceforge.dita4publishers.word2dita
- net.sourceforge.dita4publishers.xmldomain.doctypes
- net.sourceforge.dita4publishers.xmldomain.fo
- net.sourceforge.dita4publishers.xmldomain.html

4. Öffnen Sie das Toolkit auf Kommandoebene und geben Sie den Befehl `ant -f integrator.xml` ein.



```

C:\DITA-OT
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\DITA-OT1.5.2>ant -f integrator.xml
Buildfile: integrator.xml

integrate:
[integrate] Using XERCES.

BUILD SUCCESSFUL
Total time: 2 seconds
C:\DITA-OT1.5.2>

```

Abb. 2: Starten der `startcmd.bat` und Skript ausführen

5. Sie können die Generierung einer ePUB-Datei testen, in dem Sie die vom DITA-Open Toolkit mitgelieferten Doku-Quelldateien als ePUB generieren. Öffnen Sie dazu die Datei `readme.txt` im DITA-OT-Ordner `\plugins\net.sourceforge.dita4publishers.epub`. Kopieren Sie den ersten Java-Befehl `java -jar lib\dost.jar /i:doc\DITA-readme.ditamap /transtype:epub` und kopieren Sie den Befehl in das bereits geöffnete DITA-OT-Kommandofenster. Anschließend setzen Sie den Befehl durch Drücken der [Enter]-Taste ab.
- Beim DITA-OT 1.5.3 hat sich der Dateiname für die Doku-Quelldatei geändert. Sie müssen den Java-Befehl zum Generieren der ePub-Version ändern in: `java -jar lib\dost.jar /i:doc\readme.ditamap /transtype:epub`
- Im Ordner `\out` sollte sich die generierte ePUB-Datei `DITA-readme.epub` befinden.

## Links zum Artikel

1. [Dita4Publishers bei Sourceforge downloaden](#)

## Einbinden der DTDs in XMLSpy

Anleitung: So binden Sie die DTDs für pubmap, article etc in den XML-Editor XMLSpy ein, so dass Sie diese lokal validieren und benutzen können.

Bekanntlich führen mehrere Wege nach Rom. Nachfolgend zeige ich Ihnen meinen Weg, an dessen Ende Sie die spezifischen Document Type Definitions für pubmap und article usw. im Programm XMLSpy nutzen können.

1. Nennen Sie Ihren Ordner `altova\common2011\schemas\dita\files` um in `\files_original`.
2. Legen Sie einen neuen Ordner `\files` an.
3. Kopieren Sie im DITA-OT-Ordner die Ordner `\dtd\base` und `\dtd\technicalContent` und fügen diese in den soeben neu erstellten Ordner `\altova\common2011\schemas\dita\files` ein.
4. Öffnen Sie die Datei `\altova\common2011\schemas\dita\catalog.xml`. Schauen Sie, ob jede `publicId` doppelt vorkommt. Einmal ohne DITA-Versionsnummer und einmal mit DITA 1.2-Versionierung.

```
<public publicId="-//OASIS//DTD DITA Concept//EN"
uri="files/technicalContent/dtd/concept.dtd"/>
<public publicId="-//OASIS//DTD DITA 1.2 Concept//EN"
uri="files/technicalContent/dtd/concept.dtd"/>
```

5. Ändern Sie bei sämtlichen Einträgen den Pfad zu den oben kopierten DTD-Dateien im Attribut `uri` wie oben am Beispiel der `concept.dtd` gezeigt.
6. Editieren Sie in Ihrem Ordner `Dokumente` und `Einstellungen` unter `Ihren Eigenen Dateien` die Datei `\Altova\XMLSpy2011\CustomCatalog.xml`. Fügen Sie vor dem Endtag `</catalog>` folgende Zeile ein.

```
<netxCatalog catalog="C:/DITA-OT1.5.3/catalog-dita.xml"></nextCatalog.xml>
```

7. Bis Version 0.9.16 müssen Sie eventuell in die Datei `\plugins\net.sourceforge.dita4publishers.doctypes\doctypes\catalog.xml` folgenden Eintrag hinzufügen.


```
<nextCatalog catalog="mathml2/catalog.xml"/>
```

### Links zum Artikel

- [Catalogs in XMLSpy](#)

## Einbinden der Dita4Publishers-DTDs in XMLMind

Anleitung: So binden Sie die DTD für article in den XML-Editor XMLMind ein, so dass Sie diese lokal validieren und benutzen können.

-  **Hinweis:** Ich war bisher ein großer Fan des Programms, da es den `conref`-Mechanismus für den Nutzer sehr einfach werden läßt. Jedoch für Spezialisierungen, wie es die `Dita4Publishers`-DTDs sind, ist das Programm weniger gut geeignet. Mir ist es nur gelungen, die `article.dtd` einzubinden. Die Einbindung der `pubmap.dtd` ist mir nach einer Stunde nicht gelungen, so dass ich beschlossen habe, das Programm für das `Dita4Publisher`-Plugin nicht mehr zu nutzen. Vielleicht ist es ja in Zukunft möglich, die Datei `dita-catalog.xml` einzubinden. Bisher ist das nicht der Fall. Im Folgenden mein Weg, wenigstens die `article.dtd` in XMLMind aufrufen zu können.

1. Erstellen Sie einen Ordner `\plugins` innerhalb des XMLMind-Verzeichnisses `\addon\config\dita\schema\v1.2\`
2. Kopieren Sie die `doctypes`-Verzeichnisse aus dem `plugins`-Verzeichnis Ihres DITA-OTs in das soeben erstellte `plugins`-Verzeichnis des XMLMind-Programms, so dass zwei neue Verzeichnisse entstehen:

- ..\addon\config\dita\schema\v1.2\plugins\net.sourceforge.dita4publishers.doctypes
- ..\addon\config\dita\schema\v1.2\plugins\net.sourceforge.dita4publishers.xmldomain.doctypes

- Öffnen Sie die Datei ..\XMLmind\_XML\_Editor\addon\config\dita\catalog.xml und fügen Sie für die Topic Domain *article* folgende Zeilen ein:

```
<group
xml:base="schema/v1.2/plugins/net.sourceforge.dita4publishers.doctypes/doctypes/article/dtd/">
</group>
```

- Öffnen Sie die Datei *catalog.xml* im Verzeichnis ..\plugins\net.sourceforge.dita4publishers.doctypes\doctypes\article\dtd und kopieren Sie die 4 Einträge zwischen dem Catalog-Tag und fügen Sie diese zwischen den im Schritt 3 erstellten Tag *group* ein.

```
<group
xml:base="schema/v1.2/plugins/net.sourceforge.dita4publishers.doctypes/doctypes/article/dtd/">
<public
publicId="urn:pubid:dita4publishers.sourceforge.net:doctypes:dita:article"
uri="article.dtd"/>
<system
systemId="urn:pubid:dita4publishers.sourceforge.net:doctypes:dita:article.dtd"
uri="article.dtd"/>
<public publicId="urn:pubid:dita4publishers.sourceforge.net:modules:article"
uri="article.mod"/>
<system systemId="urn:pubid::modules:article.mod" uri="article.mod"/>
</group>
```

- Wiederholen Sie die Schritte 3 und 4 für alle Domains, die Sie nutzen möchten. Mindestens jedoch noch die für *common*.
- Öffnen Sie die Datei *topic.xxe* im XMLMind-Ordner *addon\config\dita* und fügen Sie einen Eintrag *rootElementLocalName* für *article* hinzu. Wiederholen Sie dies für weitere Domains.

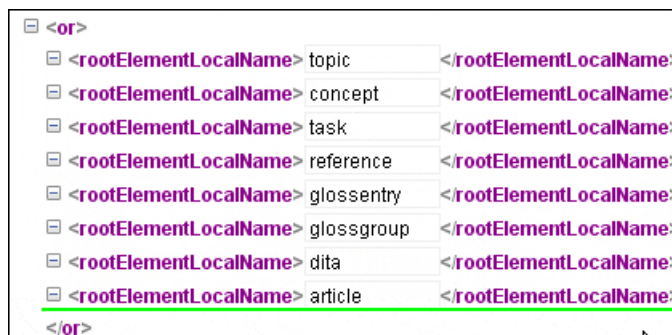


Abb. 3: *topic.xxe* editieren

- Öffnen Sie die Datei *topic.css* im XMLMind-Verzeichnis *\addon\config\dita\css* und nehmen Sie ähnliche Änderungen vor wie hier angegeben.

- Um einen Zeilenumbruch bei Subsection und Sidebar zu erhalten, kompletterieren Sie den Eintrag *display: block*:

```
example, section, subsection, sidebar
{
    display: block;
    margin: 1.33ex 0;
}
```

- Fügen Sie folgenden Eintrag ein, um die Titel bei Subsection und Sidebar zu vergrößern.

```
subsection > title,
sidebar > title
{
```

```

    font-size: 1.3em;
    margin: 1ex 0;
}

```

- c) Komplettieren Sie folgenden Eintrag um die article-Notation, um die Schriftgröße des article-Titels einzustellen.

```

/* glossentry has no title */
glossgroup > title,
concept > title,
task > title,
reference > title,
article > title,
topic > title {
    font-size: 1.5em;
    margin: 0.83ex 0;
    border-width: 1px;
    border-style: solid;
    border-top-color: transparent;
    border-left-color: transparent;
    border-right-color: transparent;
}

```

- d) Komplettieren Sie folgenden Eintrag um die article-Notation, um die Schriftfarbe des article-Titels einzustellen.

```

body example > title,
body section > title,
article > title,
topic > title
{
    color: #6090B0;
}

```

# ePub-Dateien erstellen

## ePub ohne CSS generieren

Anleitung: So generieren Sie im Plugin Dita4Publishers schnell eine ePub-Datei ohne individuelle CSS-Datei.

1. Im Ordner des ePUP-Plugins `C:\DITA-OT1.5.2\plugins\net.sourceforge.dita4publishers.epub` befindet sich eine Datei `readme.txt`. Darin sind zwei build-Befehle zur Generierung der im Toolkit befindlichen DITA-Dokumentation aufgeführt:
  - `java -jar lib\dost.jar /i:doc\DITA-readme.ditamap /transtype:epub`
  - `java -jar lib\dost.jar /i:doc\userguide\DITA-userguide.ditamap /transtype:epub`

Notieren Sie sich analog dieses Beispiels einen eigenen java-Build-Befehl, in dem Sie den Pfad für die Input-Datei nach dem Schalter `/i`: entsprechend Ihrer Input-Datei anpassen.
2. Doppelklicken Sie auf die Datei `startcmd.bat` (Windows) innerhalb Ihres DITA-OT-Ordners, um ein Kommandofenster zur Eingabe von build-Befehlen zu öffnen.
3. Geben Sie Ihren soeben angepaßten build-Befehl ein und drücken Sie [ENTER].

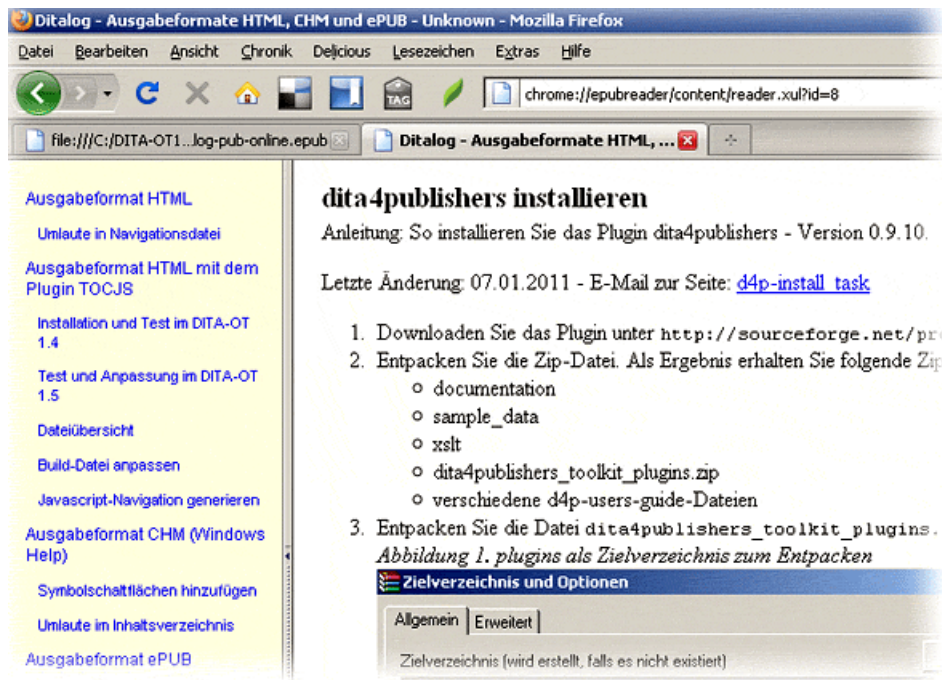


Abb. 4: Darstellung des Outputs im Firefox-Browser

Da kein Schalter `/css`: gesetzt wurde, ist die Darstellung recht spartanisch.

### Links zum Artikel

- [Firefox Addon ePub-Reader](#)

## ePub über Java generieren

Anleitung: So generieren Sie im Plugin Dita4Publishers eine ePub-Datei mit einer individuellen CSS-Datei mit Hilfe eines Java-Kommandos.

1. Im Ordner des ePUP-Plugins `C:\DITA-OT1.5.2\plugins\net.sourceforge.dita4publishers.epub` befindet sich eine Datei `readme.txt`. Darin sind zwei build-Befehle zur Generierung der im Toolkit befindlichen DITA-Dokumentation aufgeführt:

- `java -jar lib\dost.jar /i:doc\DITA-readme.ditamap /transtype:epub`
- `java -jar lib\dost.jar /i:doc\userguide\DITA-userguide.ditamap /transtype:epub`

Notieren Sie sich analog dieses Beispiels einen eigenen java-Build-Befehl, in dem Sie den Pfad für die Input-Datei nach dem Schalter `/i`: entsprechend Ihrer Input-Datei anpassen.

2. Fügen Sie dem Befehl einen Schalter für die CSS-Datei hinzu. Dieser könnte bspw. so aussehen:

```
/css:C:/DITA-OT1.5.2/css/meinstylesheet.css
```

3. Doppelklicken Sie auf die Datei `startcmd.bat` (Windows) innerhalb Ihres DITA-OT-Ordners, um ein Kommandofenster zur Eingabe von build-Befehlen zu öffnen.
4. Optional: Informieren Sie sich mit dem Hilfeschalter `-h`, ob Sie vielleicht weitere Angaben benötigen.

```

DITA-OT
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\DITA-OT1.5.2>java -jar lib\dost.jar -h
Using XERCEs in SUN JDK 1.5
java -jar lib\dost.jar [mandatory parameters] [options]
Mandatory parameters:
  /i:                specify path and name of the input f
  /transtype:       specify the transformation type
Options:
  -help, -h         print this message
  -version          print the version information and ex
  /basedir:         specify the working directory
  /ditadir:         specify the toolkit's home directory

  /outdir:          specify the output directory
  /tempdir:         specify the temporary directory
  /logdir:          specify the log directory
  /ditaext:         specify the file extension name to b
directory. Default is ".xml"
  /filter:          specify the name of the file that co
laggin/revision information
  /draft:           specify whether to output draft info
"no" and "yes". Default is "no" (hide them).
  /artlbl:         specify whether to output artwork fi
es are "no" and "yes"
  /ftr:            specify the file to be placed in the
ng area
  /hdr:            specify the file to be placed in the
ng area
  /hdf:            specify the file to be placed in the
  /csspath:        specify the path for css reference
  /css:            specify user css file
  /cssroot:        specify the root directory for user
  /copycss:        specify whether to copy user specifi
values are "no" and "yes"

```

Abb. 5: Hilfe zur `dost.jar` aufrufen

5. Geben Sie Ihren build-Befehl im Kommandofenster des DITA-OTs ein. Er könnte beispielsweise so aussehen:

```
java -jar lib\dost.jar /i:C:/DITA-OT1.5.2/ditalog.ditamap
/transtype:epub /filter:C:/DITA-OT1.5.2/filters/ditalog.ditaval
/css:C:/DITA-OT1.5.2/css/ditalog.css /outdir:C:/DITA-OT1.5.2/out/epub-ditalog
/logdir:C:/DITA-OT1.5.2/log
```

Ersetzen Sie den Begriff `ditalog` durch Ihre Dateinamen.

- Im oben angegebenen Outputverzeichnis `/outdir`: wurde eine ePUB-Datei generiert. Sie stellt sich im Firefox-Browser mit individueller CSS wie folgt dar:

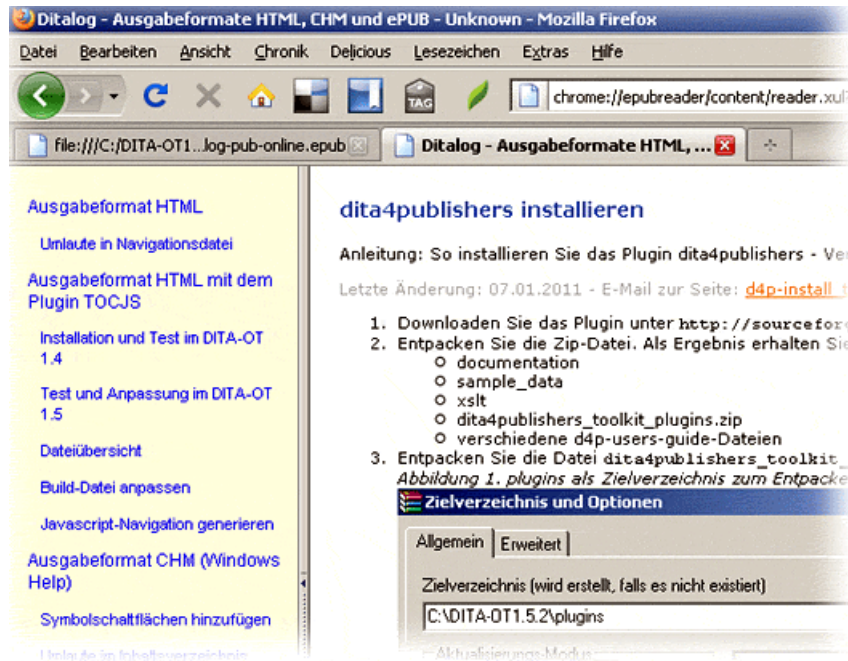


Abb. 6: ePub mit CSS

### Links zum Artikel

- [Firefox Addon ePub-Reader](#)

## ePub über eine Build-Datei generieren

Anleitung: So generieren Sie im Plugin Dita4Publishers eine ePub-Datei mit einer individuellen CSS-Datei mithilfe einer Ant-Datei.

- Nehmen Sie gegebenenfalls Änderungen in der Datei `build_transtype-epub_template.xml` im Ordner `C:\DITA-OT1.5.2\plugins\net.sourceforge.dita4publishers.epub` vor. Ändern Sie z.B. den Eintrag:
 

```
<property name="idURISTub" value="http://IhreDomain.com/" />
```
- Erstellen Sie eine ePUB-Build-xml-Datei. Im D4P-User-Guide heißt der wichtigste Satz, der von jedem Nutzer mühsam entdeckt werden muss, folgendermaßen:
 

*You can use the EPUB transformation type by simply specifying a transtype value of "epub" rather than "html" and use the same parameters you would use for HTML generation.*

In meinem Fall hieß das, dass ich meine bisherige Build-Datei für die HTML-Generierung nach dem Eintragen des transtypes `epub` soweit zusammenstrich, bis die ePUB-Generierung fehlerlos durchlief. Meine build-Datei `build-epub.xml` sieht folgendermaßen aus:

```
<project basedir="..\.." default="all" name="ditalog">
<property name="dita.dir" value="C:${file.separator}DITA-OT1.5.2" />
<import file="${dita.dir}/integrator.xml" />
<import file="${dita.dir}\build.xml" />
<target name="all" depends="integrate" >
<ant antfile="${dita.dir}/build.xml" target="init" >
```

```

<property name="args.logdir" value="\${dita.dir}\${file.separator}log" />
<property name="args.input"
value="\${dita.dir}\${file.separator}ditalog.ditamap" />
<property name="output.dir"
value="\${dita.dir}\${file.separator}out\${file.separator}epub-ditalog" />
<property name="transtype" value="epub" />
<property name="args.css"
value="\${dita.dir}\${file.separator}css\${file.separator}ditalog.css" />
<property name="args.csspath" value="css" />
<property name="args.dita.locale" value="de-de" />
<property name="args.copycss" value="yes" />
<property name="dita.input.valfile"
value="\${dita.dir}\${file.separator}filters\${file.separator}ditalog.ditaval"
/>
<property name="validate" value="true" />
</ant>
</target>
</project>

```

3. Doppelklicken Sie auf die Datei `startcmd.bat` (Windows) innerhalb Ihres DITA-OT-Ordners, um ein Kommandofenster zur Eingabe von build-Befehlen zu öffnen.
4. Geben Sie den `ant`-Befehl ein und drücken Sie auf [ENTER]. Im Beispiel ist dies:

```
ant -f C:\DITA-OT1.5.2\build\build-epub.xml -logger
org.dita.dost.log.DITAOTBuildLogger
```

5. Im `output.dir` befindet sich eine ePUP-Datei. Im Beispiel ist das die Datei `ditalog.epub`. Bitte setzen Sie für den Begriff *Ditalog* Ihre individuellen Dateinamen.

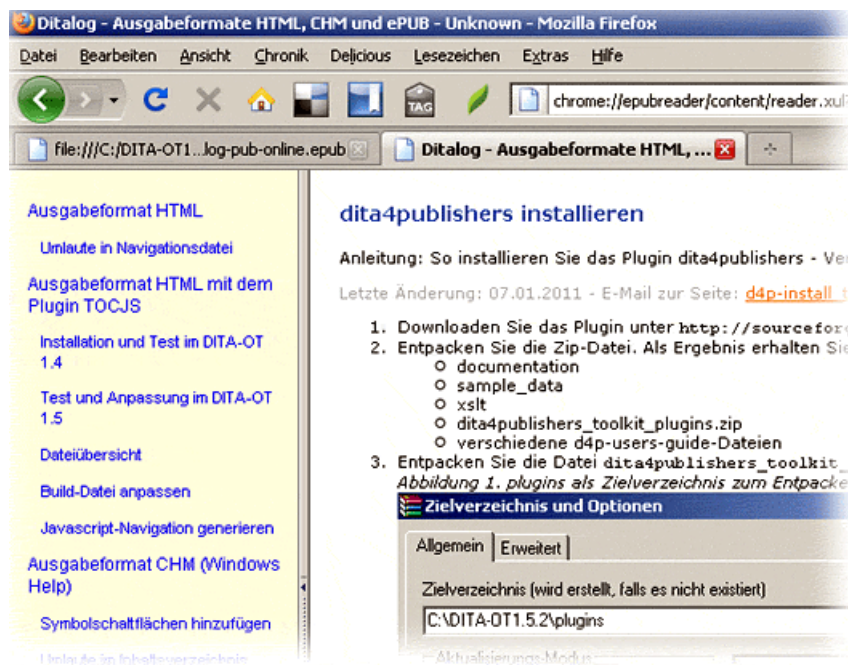


Abb. 7: ePub mit CSS

## Links zum Artikel

- [Firefox Addon ePub-Reader](#)

## Autor unbekannt in ePub-Vorschau

Geben Sie Autor und Verleger in einer Dita-map an, so dass in der *Adobe Digital Editions* bei Autor und Verleger Ihre Angaben stehen.

### Fehlende Angaben in Adobe Digital Editions



Abb. 8: Unbekannter Autor und Verlag

In vielen ePUB-Reader fällt der Umstand der fehlenden Angaben zu *Autor* und *Verlag* nicht auf. So z.B. im Firefox-Addon *ePUB-Reader*. Jedoch in *Adobe Digital Editions* wirken die *unbekannt*-Vermerke wenig vertrauenserweckend.

### Angaben in der ditamap hinterlegen

Hinterlegen Sie in Ihrer Dita-map-Datei nach dem `map`-Tag und vor dem ersten `topicref`-Tag die `topicmeta`-Angaben wie in diesem Beispiel.

```
<map id="DDR-Protagonisten" scope="local" title="DDR-Protagonisten der 70er und
80er Jahre" xml:lang="de-de">
<topicmeta>
<author>Andreas Petersell</author>
<publisher>Andreas Petersell</publisher>
<copyright>
<copyyear year="2011"></copyyear>
<copyrholder>Andreas Petersell</copyrholder>
</copyright>
</topicmeta>
...
```

### Links zum Artikel

1. [Adobe Digital Editions](#)

## Related Links abstellen

So stellen Sie die Navigationslinks und die Links zur Übersicht ab.

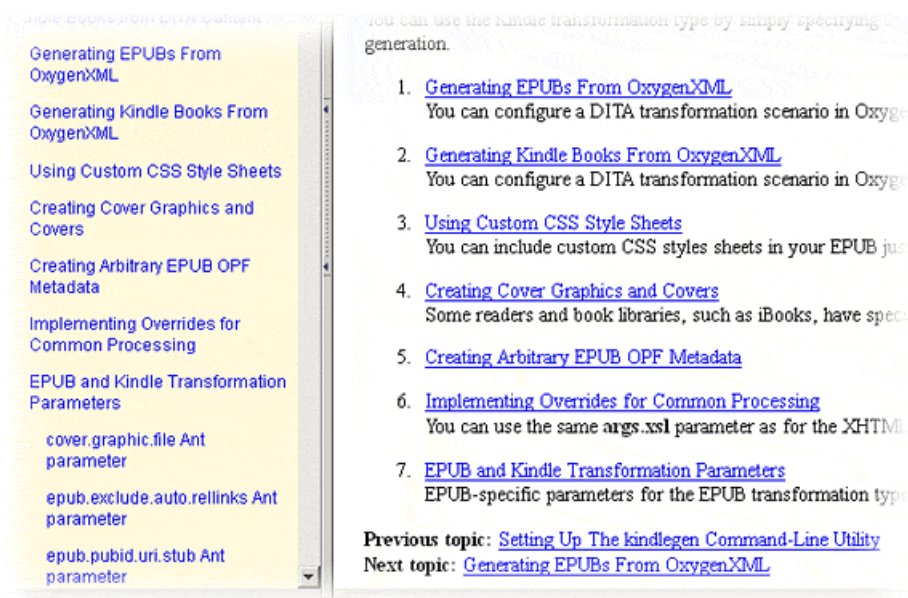


Abb. 9: Verzichtbare Links im ePub

Da ich obige Links im ePub nie gebrauchen werden, habe ich sie dauerhaft abgestellt. Danach sollten die automatisch erstellten Überblick-Links (1-7) und die *Weiter-* und *Zurück*-Links nicht mehr zur Anzeigen kommen.

Öffnen Sie die beiden Dateien im Dita4Publishers-Plugin:

`\net.sourceforge.dita4publishers.epub\build_transtype-epub.xml` und `build_transtype-epub_template.xml`. Fügen Sie folgenden Eintrag hinzu:

```
<property name="epub.exclude.auto.rellinks" value="true"></property>
```

## Navigation eindeutsch

Bei Gebrauch einer pubmap soll statt *Chapter* und *Part Kapitel* und *Teil* im Inhaltsverzeichnis stehen.

Öffnen Sie folgende 2 Dateien im Dita4Publisher-Plugin:

`\net.sourceforge.dita4publishers.epub\xsl\map2epubD4Pimpl.xml` und `map2epubBookmapImpl.xml` und ersetzen Sie die Begriffen *chapter* durch *Kapitel* und *Part* durch *Teil*:

```
<xsl:text>Teil </xsl:text><!-- FIXME: Enable localization of the string. -->
```

```
<xsl:text>Kapitel </xsl:text><!-- FIXME: Enable localization of the string. -->
```

## Covergraphik in Standard-Ditamap

Anleitung: So binden Sie in einer ditamap eine Covergraphik ein.

Voraussetzung: Sie haben eine Covergraphik-Datei in der Größe 600 x 900px parat.



**Hinweis:** In der Datei

`/plugins/net.sourceforge.dita4publishers.epub/build_transtype-epub.xml` muss der Eintrag ohne Value vorhanden sein: `<property name="cover.graphic.file" value=""></property>`.

1. Erstellen Sie innerhalb ihrer Topic-XML-Dateien eine Ordner `\covers\images` und kopieren Sie die Covergraphik-Datei in diesen Ordner.

2. Fügen Sie Ihrer ditamap-Datei nach dem title-Tag ein Tag `topicmeta` mit einem data-Element ein. Dies könnte folgendermaßen aussehen:

```
<topicmeta>
<author>Andreas Petersell</author>
...
<data name="covergraphic">topics/ddr/covers/images/cover-ddrprotag.jpg</data>
</topicmeta>
```

Sie haben in Ihrer ePub-Datei ein Coverbild eingefügt, das nicht auf einer normalen Seite mit ausgegeben wird. Es dient in vielen Programmen lediglich als Cover, wie hier im Programm Calibre 0.8.7

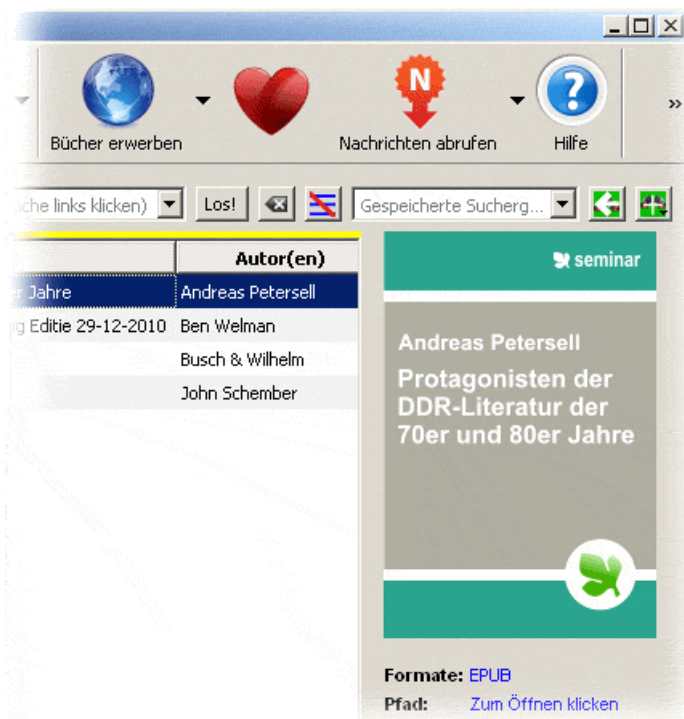


Abb. 10: Anzeige des Covers in Calibre

Nachtrag vom 09.11.2011 - leider funktioniert dieser Weg nur mit dem Java-Befehl `java -jar lib\dost.jar`  
 . . . . Vielleicht muss noch das DITA-OT aktuell gehalten werden und das neueste D4P-Plugin eingespielt werden?

# D4P-Vokabular

---

So arbeiten Sie mit den Dita4Publishers-Spezialisierungen und Formatierungsmöglichkeiten.

## D4P-Formatierungen in Concepts nutzen

---

So binden Sie die Dita4Publishers-Formatierungsmöglichkeiten in die DITA-Standard-Dateitypen concept, task, reference, topic und map ein.

Nach meiner erstmaligen Installation des Plugins hatte ich mich gefragt, warum der Autor die Standard-DTDs von **Concept** u.a. in den `doctypes`-Ordner mit aufführt. Neulich war ich innerhalb eines Concepts in der Not, einen Zeilenumbruch via `</br>` zu haben. Erst jetzt erinnerte ich mich der Standard-DTDs im Plugin-Verzeichnis. Kaum hatte ich anstelle der Standard-Concept-DTD auf die Plugin-DTD im Public Identifier verwiesen, erschienen mir auch im Concept sämtliche D4P-Domains wie z.B. `<br>`.

### Ändern der Dokumenttypendeclaration

Ersetzen Sie innerhalb Ihrer Concept-Datei den alten Eintrag, z.B.

```
<!DOCTYPE concept PUBLIC "-//OASIS//DTD DITA Concept//EN"
"../../../../dtd/technicalContent/dtd/concept.dtd">
```

durch den neuen D4P-Identifizier

```
<!DOCTYPE concept PUBLIC
"urn:pubid:dita4publishers.sourceforge.net:doctypes:dita:concept"
"../../../../plugins/net.sourceforge.dita4publishers.doctypes/doctypes/concept/dtd/concept.dtd">
```

Den Dateipfad, auf den im Fehlerfall des Public Identifiers zurückgegriffen wird, müssen Sie anpassen, oder evtl. weglassen.

### Links zum Artikel

1. [Wikipedia: Dokumenttypdeklaration](#)

# Index

## A

Adobe Digital Editions 12  
article.dtd 5  
Autolinks abstellen 12  
Autor unbekannt 12

## B

build\_transtype-epub\_template.xml 10  
build-Befehl für ePUB 8, 9

## C

catalog-dita.xml 5  
Chapter 13  
chapter.dtd 5  
Cover 13  
Covergraphik 13

## D

data 13  
Dita4Publishers 3  
    installieren 3  
dost.jar 9

## E

ePUB build-Datei 10  
ePUB generieren 8, 9  
ePUB mit CSS 9  
epub.exclude.auto.rellinks 12  
ePUP-Reader Firefox Addon 12

## F

Formatierungsmöglichkeiten von D4P 15

## P

Part 13  
part.dtd 5  
pubmap.dtd 5

## R

related links 12

## S

sidebar.dtd 5  
subsection.dtd 5

## T

topic.css 5  
topic.xxe 5  
topicmeta 12, 13

## V

Verlag unbekannt 12

## X

XMLMind 5  
XMLSpy 5

## Z

Zeilenumbruch 15