



Ditalog - Ausgabeformat PDF

Über den Autor.....	2
Impressum.....	2
Formatierung und Design.....	3
A4 als Standardseitenformat.....	3
Metadaten der PDF-Datei erzeugen.....	4
Einzelkapitel in einer Bookmap.....	5
Texteinrückung mit hängender erster Zeile.....	6
Fußzeile ausgeben.....	7
Fußzeile im Template hinterlegen.....	8
Paragraphensymbol § im indexterm.....	8
Online versus Druckausgabe.....	10
Inhaltsverzeichnis Print vs. Online.....	10
xref-Direktlinks in Onlinehilfe vs. Print.....	11
PDF in der Technischen Dokumentation.....	13
Zugang zum Bildschirm voranstellen.....	13
Zugangsweg im PDF-Output darstellen.....	13
Weitere Tipps und Werkzeuge.....	16
Installation des RenderX-Formatierers.....	16
Umzug des RenderX-Formatierers.....	18
Leere Seiten aus der PDF-Datei entfernen.....	19

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
Ausgabe:	Ditalog - Ausgabeformat PDF
Redaktionsschluß:	28.11.2011

Formatierung und Design

A4 als Standardseitenformat

A4 ist für Europa ein Standard. Nicht jedoch für die Entwickler des Idiom FO Plugins. Es gilt, das Standardseitenformat A4 für den PDF-Output einzustellen.

Egozentrik des Druckers

Ich sitze (sic!) viel im Büro. Da kam es mir schon gelegen, dass mein Drucker mich persönlich sehen wollte: immer dann, wenn ein PDF-Druck aus dem DITA-OT mit Hilfe des Idiom FO-Plugins anstand. Immer fragte mich der Drucker via Display, ob er auf A4 umstellen solle. Nun habe ich Dank eines Tutorials auf Scriptorium meine gesundheitsfördernden Spaziergänge (2 m) zum Drucker abgestellt.

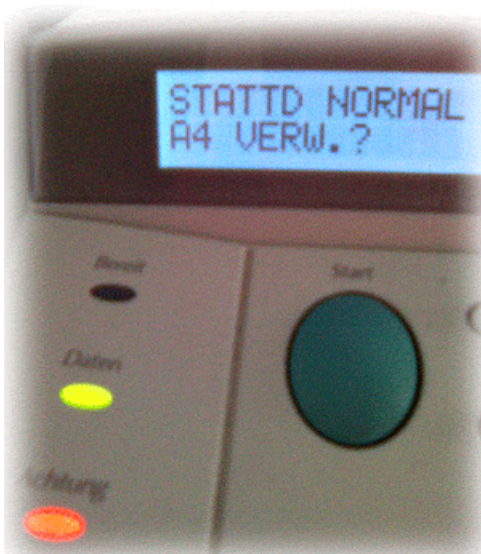


Abb. 1: Kein sofortiger Ausdruck

A4 beim DITA-OT 1.4.2.1

Editieren Sie die Datei `fo\cfg\fo\layout-master.xml` und tragen Sie die folgenden Werte in mm für A4 bei sämtlichen Seitenbreiten und Seitenhöhen ein:

```
page-width="210mm"
page-height="297mm"
```

A4 beim DITA-OT 1.5

Editieren Sie die Datei `fo\cfg\fo\attrs\basic-settings.xml` und tragen Sie die Werte in mm für A4 ein:

```
<xsl:variable name="page-width">210mm</xsl:variable>
<xsl:variable name="page-height">297mm</xsl:variable>
```

Links zum Artikel

1. [Scriptorium.com](http://www.renderx.com/) - *Customizing PDF output in the DITA Open Toolkit*

Metadaten der PDF-Datei erzeugen

So generieren Sie über das DITA OT und dem FO-Plugin die Metadaten einer PDF-Datei.

Fehlenden Metaangaben

Auf zweierlei Wegen kann man sich die Eigenschaften einer PDF-Datei anzeigen lassen. Im Acrobat Reader unter dem Menüpunkt **Datei > Eigenschaften**. Im Windows-Explorer per rechten Mausklick und Menüeintrag **Eigenschaften**. Nun wies mich Ben (www.softwijs.nl), der seine Anleitungen ebenfalls mit DITA realisiert, darauf hin, dass in meinen PDFs die Meta-Angaben fehlen. Kurz darauf gab es zufällig einen Forum-Eintrag in der *DITA-User-Yahoogruppe* bezüglich der Meta-Daten in einer PDF-Datei. Hier die Umsetzung des Forum-Beitrages.

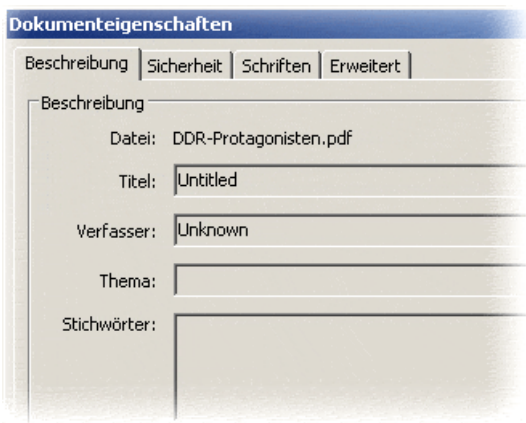


Abb. 2: Fehlende Angaben hier im Acrobat Reader

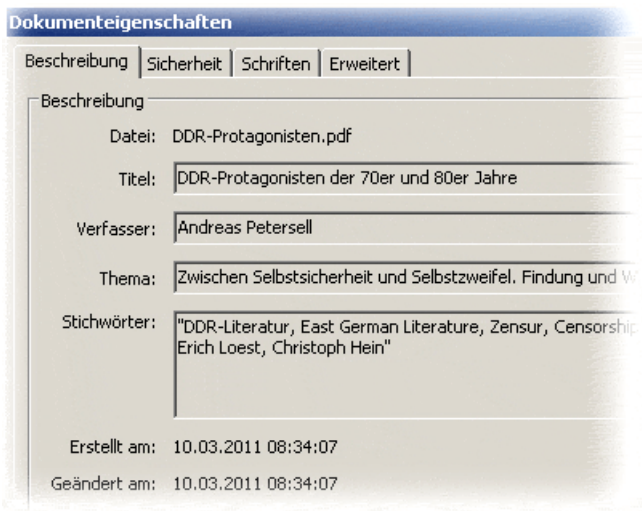


Abb. 3: So sollte es mit Meta-Angaben aussehen

Bearbeiten der root-processing.xml

Öffnen Sie die Datei `DITA-OT\demo\fo\xsl\fo\root-processing.xml`. Fügen Sie gleich nach dem Eintrag `<fo:root xsl:use-attribute-sets="__fo__root">` folgendes ein:

```
<rx:meta-info>
<rx:meta-field name="author" value="_____" />
<rx:meta-field name="creator" value="_____" />
<rx:meta-field name="title" value="_____" />
```

```
<rx:meta-field name="subject" value="_____" />
<rx:meta-field name="keywords" value="_____" />
<rx:meta-field name="publisher" value="_____" />
</rx:meta-info>
```

Ersetzen Sie die Unterstriche bei **value** jeweils mit Ihren Angaben. Damit der Renderx-Präfix `rx:` validiert werden kann, muss, wenn noch nicht vorhanden, folgender Eintrag für `rx:` in die Stylesheet-Deklaration hinzugefügt werden:

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fo="http://www.w3.org/1999/XSL/Format"
  xmlns:rx="http://www.renderx.com/XSL/Extensions"
  version="2.0">
```

Der Autor des Yahoo-Group-Beitrages betont, dass die statischen Angaben, die er unter `value` getätigt hat, sich bestimmt auch durch Variablen aus den Bookmap-Metadaten füllen lassen könnten. Doch das ist ein Thema für sich.

Links zum Artikel

1. [Softwijs Anleitungen](#)
2. [Yahoo-Groups-Post zu Metaangaben im PDF](#)

Einzelkapitel in einer Bookmap

Es sollen Einzelkapitel und Kapitel mit Unterabschnitten im Inhaltsverzeichnis einer bookmap korrekt dargestellt werden.

So soll es aussehen

Kapitel 2: Überblick zur DDR-Literatur.....	9
Kapitel 3: Die Protagonisten bei Loest, Maron, Becker und Hein.....	19
Erich Loest: Es geht seinen Gang oder Mühen in unserer Ebene.....	20
Jurek Becker: Schlaflose Tage.....	22
Monika Maron: Flugasche.....	23
Christoph Hein: Der fremde Freund.....	24
Kapitel 4: Schlussbemerkung.....	27

Abb. 4: Einzelkapitel in 2. und 4.

Damit Kapitel 2 und 4 im Plugin nicht auf der typischen zweigeteilten Startseite eines Kapitels landen, ist einiges in der bookmap zu beachten.

Einzelkapitel in der bookmap-Datei

Für die fettgedruckte Kapitelüberschrift im PDF-Output muss immer eine Extra-Datei erstellt werden. So existieren für das *Kapitel 2* zwei Dateien. Die Topic-Datei für die Kapitelüberschrift und eine Concept-Datei für den eigentlichen Kapiteltext. Topic und Concept sind in der Bookmap folgendermaßen eingebunden.

```
<chapter href="topics/ddr/chapters/2_ueberblick_topic.xml"
  id="auswahl_topic" locktitle="yes"
  navtitle="Überblick zur DDR-Literatur" scope="local" toc="yes"
  type="topic" xml:lang="de-de">
  <topicref href="topics/ddr/chapters/2_ueberblick_con.xml" id="auswahl_con"
  scope="local" toc="no" type="concept" xml:lang="de-de"/>
</chapter>
```

Wichtig im Tag `<chapter>` die Attribute `locktitle` und `navtitle` sowie im `<topicref>` das Attribut `toc`.

Kapitel mit Unterabschnitten in der bookmark-Datei

Es muss eine Datei für die Kapitelüberschrift sowie die entsprechende Anzahl an Dateien für die Unterabschnitte erstellt werden.

```
<chapter href="topics/ddr/chapters/3_protagonisten_topic.xml"
id="protagonisten_topic" scope="local" toc="yes" type="topic" xml:lang="de-de">

  <topicref href="topics/ddr/chapters/3-1_loest_con.xml" id="loest_con"
scope="local" toc="yes" type="concept" xml:lang="de-de"/>


  <HIER DIE 3 WEITEREN TOPICREFS>

</chapter>
```

Wichtig hier im `<topicref>` ist das Attribut `toc="yes"`.

Texteinrückung mit hängender erster Zeile

Anleitung: So erstellen Sie einen Literaturverzeichniseintrag mit hängender 1. Zeile.

 **Hinweis:** Der PDF-Output wird mit Hilfe des Idiom-FO-Plugins realisiert. Letzteres verwendet den RenderX als XSL-FO-Prozessor. Die "Personal Edition" unter www.renderx.com/download/personal.html ist kostenlos und zeitlich unbegrenzt für nichtkommerzielle Zwecke nutzbar.

1. Für ein Literaturverzeichnis sollen die Absätze pro Eintrag eingerückt werden mit der ersten Zeile hängend.

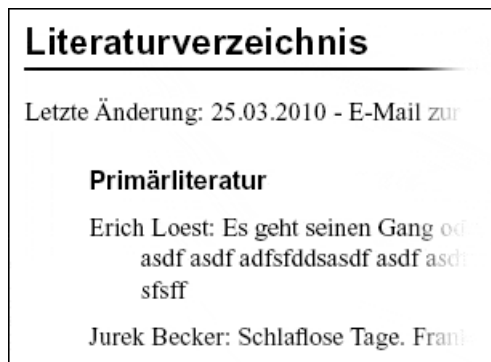


Abb. 5: Hängende erste Zeile

2. Fügen Sie in der Datei `custom.xsl` im Ordner `\demo\fo\Customization\fo\xsl` folgenden Eintrag für ein Template hinzu:

```
<xsl:template match="*[contains(@class, ' topic/p ')]">
<xsl:choose>
<xsl:when test="@otherprops='indent'">
<fo:block text-indent="-2em" start-indent="4.5em">
<xsl:apply-templates/>
</fo:block>
</xsl:when>
<xsl:otherwise>
<fo:block xsl:use-attribute-sets="p" id="{@id}">
<xsl:apply-templates/>
</fo:block>
</xsl:otherwise>
</xsl:choose>
</xsl:template>
```

Neu ist hier die `when`-Anweisung. Die `otherwise`-Anweisung ist das normale Template für einen Absatz. Für indent können Sie eigene Entsprechungen finden. Es soll hier nur als Beispiel dienen.

3. Versehen Sie sämtliche Absätze, die Quellenangaben enthalten sollen, mit dem Attribut `otherprops="indent"`.

```
<p otherprops="indent" outputclass="indent">
Erich Loest: Es geht seinen Gang oder Mühen in unserer Ebene. München 1994
(dtv 10430)</p>
```

4. Optional: Falls Sie die Texteinrückung auch in XHTML-Output haben möchten, vergeben Sie wie oben ein Attribut `outputclass="indent"`. Erstellen Sie anschließend in der CSS-Datei eine Deklaration gleichen Namens mit einem Eintrag für `text-indent`.

```
.indent
{
text-indent: -2em;
margin-left: 2em;
}
```

Fußzeile ausgeben

Anleitung: So geben Sie eine einfache, zentrierte Fußzeile aus.

1. Fügen Sie in der Datei `DITA-OT\demo\fo\Customization\fo\attr\custom.xsl` folgende zwei Attribut-Sets hinzu:

```
<xsl:attribute-set name="__body__odd__footer">
  <xsl:attribute name="text-align">center</xsl:attribute>
  <xsl:attribute name="margin-bottom">10pt</xsl:attribute>
</xsl:attribute-set>
```

```
<xsl:attribute-set name="__body__even__footer">
  <xsl:attribute name="text-align">center</xsl:attribute>
  <xsl:attribute name="margin-bottom">10pt</xsl:attribute>
</xsl:attribute-set>
```

2. Füllen Sie die Variablen *Body odd footer* und *Body even footer* in der Datei `DITA-OT\demo\fo\Customization\common\vars\de_DE.xml`. So könnte dies beispielhaft aussehen.

```
<variable id="Body odd footer">
&#169; &#169; 2010 &#169; Andreas &#169; Petersell &#169; - &#169; www.ditalog.com
</variable>
```

Füllen Sie bei Bedarf ebenfalls die Variable *Body even footer*.

© 2010 Andreas Petersell - www.ditalog.com

XML to PDF by RenderX XEP XSL-FO Formatter, visit us at <http://www.renderx.com/>

<http://www.ditalog.com>


Ditalog 1.0 | Version

Abb. 6: Fußzeile - bei Internetdomain wird die Linkfähigkeit erhalten

Der Vorteil dieser Vorgehensweise ist die Schnelligkeit des Umsetzens. Ein gravierender Nachteil wird offenbar, wenn mehrere Layouts administriert werden müssen: beim Wechsel eines Layouts müssen die Variablen für die footer neu editiert werden.

Fußzeile im Template hinterlegen

Anleitung: So geben Sie eine einfache, zentrierte Fußzeile mit Hilfe eines angepaßten xsl-Templates im Customization-Ordner des FO-Plugins aus.

-  **Hinweis:** Der PDF-Output wird mit Hilfe des Idiom-FO-Plugins realisiert. Letzteres verwendet den RenderX als XSL-FO-Prozessor. Die "Personal Edition" unter www.renderx.com/download/personal.html ist kostenlos und zeitlich unbegrenzt für nichtkommerzielle Zwecke nutzbar.

1. Fügen Sie in der Datei DITA-OT\demo\fo\Customization\fo\attr\custom.xml folgende zwei Attribut-Sets hinzu:

```
<xsl:attribute-set name="__body__odd__footer">
  <xsl:attribute name="text-align">center</xsl:attribute>
  <xsl:attribute name="margin-bottom">10pt</xsl:attribute>
</xsl:attribute-set>
```

```
<xsl:attribute-set name="__body__even__footer">
  <xsl:attribute name="text-align">center</xsl:attribute>
  <xsl:attribute name="margin-bottom">10pt</xsl:attribute>
</xsl:attribute-set>
```

2. Stellen Sie sicher, dass Sie im Ordner \demo\fo\Customization\fo\xsl eine Datei namens custom.xml haben.
3. Kopieren Sie folgende zwei Templates - insertBodyOddFooter und insertBodyEvenFooter aus der Datei \demo\fo\xsl\fo\static-content.xml in oben erwähnte custom.xml.
4. Fügen Sie in beide Templates - so Sie für die geraden und ungeraden Fußzeilen einen Eintrag möchten - Ihre xsl-text-Zeile hinzu:

```
<xsl:template name="insertBodyOddFooter">
  <fo:static-content flow-name="odd-body-footer">
    <fo:block xsl:use-attribute-sets="__body__odd__footer">
      <!-- HIER FUSSZEILENEINTRAG z.B. -->
      <xsl:text>&#169;&#160;2010&#160;Andreas&#160;Petersell&#160;-&#160;www.ditalog.com</xsl:text>
      <xsl:call-template name="insertVariable">
        <xsl:with-param name="theVariableID" select="'Body odd footer'"/>
      <!-- hier der Template-Rest -->
    </fo:block>
  </fo:static-content>
</xsl:template>
```

5. Im Tag <xsl:text> steht Ihr individueller Fußzeilentext. Wiederholen Sie diesen Eintrag im Template insertBodyEvenFooter, wenn dieser Text auf geraden Seiten ebenfalls ausgegeben werden soll.

© 2010 Andreas Petersell - www.ditalog.com
 XML to PDF by RenderX XEP XSL-FO Formatter, visit us at <http://www.renderx.com/>
<http://www.ditalog.com>
 Ditalog 1.0 | Version

Abb. 7: Ausgabe der Beispiel-Fußzeile

Paragraphensymbol § im indexterm

Innerhalb des indexterm-Tags wird das Paragraphensymbol nicht ausgegeben und der Stichworteintrag entfällt als Ergebnis ganz.

Es kann sein, dass Ihre Stichworteinträge, beginnend mit dem Paraphensymbol, nicht geparkt und somit nicht im Stichwortverzeichnis auftauchen. Behelfen Sie sich mit der Nummer im ISO-8859-1-Character-Set oder dem Namen. Im Falle des Paragraphensymbols (Section Sign) ist dies `§` bzw. `§`

Zwei zwingende Voraussetzungen:

1. Texteingabe nur in der Quellcode-Ansicht des Editors. XML Mind macht aus dem `§` gnadenlos ein `&sect;` - mit dem Ergebnis, dass dieses Stichwort falsch dargestellt wird.
2. Das Paraphensymbol darf nicht zu Beginn des Stichworteintrages stehen. Setzen Sie z.B. das Kürzel des Gesetzes davor: `StAG 3 Abs. 2.`

Manchmal sind aber auch Quellcode-Editoren störrisch. Da hilft, wir bei mir, nur ein gutes Suchen-Ersetzen-Tool für das ändern von `&#167` in `§`.

Online versus Druckausgabe

Inhaltsverzeichnis Print vs. Online

Unterschiede zwischen Print- und Online-Inhaltsverzeichnis.

1:1-Übertragung des Online-Verzeichnisses?

Meistens ist immer die Online-Hilfe das erst Output-Ergebnis und man macht sich daran, die Haupt-Ditamap als Grundlage für eine Bookmap für den PDF-Output zu nehmen. So sah dann mein erstes Ergebnis des PDF-Inhaltsverzeichnisses so aus:

Einleitung Praxisbeispiele für Redakteure	
Kapitel 1: Ausgabeformate	
CHM-Format
Umlaute im Inhaltsverzeichnis
HTML-Format
Plugin tocjs
Kapitel 2: Online-Hilfe	
Kontextsensitive Hilfe per Ditamap
E-Mail-Link für Feedback
Kapitel 3: Tools	
DITA OT
DITA OT-Installation

Spätestens jetzt muss man sich Gedanken machen, ob eine 1:1-Kopie des Online-Verzeichnisses für den PDF-Output geeignet ist. Hier im Beispiel des ditalogs liegt es nahe, die verschiedenen Ausgabeformate auch unter einem Stichwort Ausgabeformate zu sammeln. Doch ist dies nicht das, was dem Leser in einem Druckdokument weiterhilft. Hier wäre es besser, den einzelnen Ausgabeformaten ein eigenes Kapitel zu widmen. Man erspart sich die dritte Ebene im Inhaltsverzeichnis und der Leser erhält einen schnelleren Überblick. Und Überblick steht bei einem Printinhaltsverzeichnis im Vordergrund. Wer wirklich etwas Spezielles sucht, muss über das Stichwortverzeichnis gehen.

Abb. 8: Erster Output im PDF-Format

Besondere Gestaltung der bookmap

Die Bookmap bedarf also meistens eine Überarbeitung und sieht im Ergebnis anders als die Online-ditamap aus.

Einleitung Praxisbeispiele für Redakteure	
Kapitel 1: HTML als Ausgabeformat.....	
Plugin tocsj.....	
Kapitel 2: CHM als Ausgabeformat.....	
Umlaute im Inhaltsverzeichnis.....	
Kapitel 3: Online-Hilfe.....	
Kontextsensitive Hilfe per Ditamap.....	
E-Mail-Link für Feedback.....	
Kapitel 4: Tools.....	
DITA OT-Installation.....	

Abb. 9: Neue Anordnung

Egal, wie man die bookmark neu anordnet - man wird sie immer flexibel den Bedürfnissen des Lesers (nicht Nutzers) anpassen müssen.

xref-Direktlinks in Onlinehilfe vs. Print

Hilfreiche xref-Links in der Onlinehilfe sollen im PDF-Output ausgeblendet werden.

Direktlinks

Manchmal möchte man auf xref-Links einfach nicht verzichten, denn in der Online-Hilfe erfüllen sie ihren Zweck, der auch durch sogenannte *related links* am Ende des Artikels nicht ersetzt werden kann. Hier der pure xref-Link und das Erscheinungsbild im Output.

```
<p>Für eine Entscheidung bedarf es neben den <xref
href="stammdaten-pflichtfelder_ref.xml"> generellen Pflichtfeldern</xref>
noch folgender drei Voraussetzungen, damit sie ans EStA gesendet werden
können:</p>
```

Voraussetzungen für eine Übermittlung

Für eine Entscheidung bedarf es neben den [generellen Pflichtfeldern](#) noch folgender drei Voraussetzungen, damit sie ans EStA gesendet werden können:

1. **Rechtsgrundlage(n)** - In manchen Fällen ist eine von mehreren Rechtsgrundlagen möglich
2. **Kriterium** innerhalb Civisos, mit dessen Hilfe die [Entscheidungsform](#) realisiert wird. Dies ist ein Kriterium, das bestimmte PFLICHT- als auch KANN-Felder. Ein Kriterium aus einer bestimmten Anzahl von Kriterien sind möglich.

Abb. 10: Online ein nützlicher Link zum Term "generelle Pflichtfelder"

Voraussetzungen für eine Übermittlung

Für eine Entscheidung bedarf es neben den [EStA-Stammdaten-Pflichtfelder](#) auf Seite 24 noch folgende drei Voraussetzungen, damit sie ans BVA EStA gesendet werden können:

1. **Rechtsgrundlage(n)** - In manchen Fällen ist eine von mehreren Rechtsgrundlagen möglich
2. **Kriterium** innerhalb Civisos, mit dessen Hilfe die [Schlüsselwerte zum EStA](#) auf Seite 24 realisiert wird. Dies ist ein Kriterium, das bestimmte PFLICHT- als auch KANN-Felder. Ein Kriterium aus einer bestimmten Anzahl von Kriterien sind möglich.

Abb. 11: Im PDF erscheint der Link mit Topic-Titel

Während man mit der jeweiligen Seiteangabe noch leben könnte, erscheint der Linktext nicht mehr. Statt dessen kommt der Topic-Titel der referenzierten Seite zur Ausgabe, was immer grammatikalische Ungeheuer entstehen läßt.

Direktlinks pro Ausgabeformat differenzen

Es soll der nützliche Online-Link erhalten bleiben, der Link innerhalb der PDF-Datei jedoch verschwinden. In der ditaval-Datei müssen - sofern sie nicht schon existieren - zwei Produktattribute geschaffen werden.

```
<prop action="exclude" att="product" val="print" />
<prop action="include" att="product" val="online" />
```

Nun können in der jeweiligen Topic-XML-Datei zwei Einträge für den Term *generellen Pflichtfeldern* vorgenommen werden. Beide bekommen ein anderes Produkt-Attribut.

```
<p>Für eine Entscheidung bedarf es neben den
<xref href="stammdaten-pflichtfelder_ref.xml" product="online">generellen
Pflichtfeldern</xref>
<keyword product="print">generellen Pflichtfeldern</keyword>
noch folgender drei Voraussetzungen, damit sie ans ESTa gesendet werden
können:</p>
```

Die Online-Verlinkung bleibt erhalten, wogegen im PDF der Link verschwindet.

Voraussetzungen für eine Übermittlung

Für eine Entscheidung bedarf es neben den generellen Pflichtfeldern noch folgender drei Voraussetzungen, damit sie ans BVA ESTa gesendet werden können:

1. **Rechtsgrundlage(n)** - In manchen Fällen ist eine von mehreren Rechtsgrundlagen maßgebend.
2. **Kriterium** innerhalb Civisos, mit dessen Hilfe die *Schlüsselwerte zum ESTa* auf Seite 10 des Kriteriums hat bestimmte PFLICHT- als auch KANN-Felder. Ein Kriterium aus einer bestimmten Gruppe

Abb. 12: Term ohne Link

PDF in der Technischen Dokumentation

Zugang zum Bildschirm voranstellen

Es soll der Zugang zum jeweiligen Bildschirm der eigentlichen Handlungsanweisung vorangestellt und somit extra aufgeführt werden.

Handbuch vs. kontextsensitiver Hilfe

Es gibt zweierlei Arten des Lesens einer Online-Hilfe: linear als Buch oder per Verknüpfung aus der Anwendung. Und immer dann, wenn letzteres geschieht, also eine Handlungsanleitung gezielt durch die kontextsensitive Hilfe geöffnet wird, bedarf es keiner Beschreibung mehr, wie man zum jeweiligen Bildschirm gelangt ist.

Dies ist nur bei Online-Hilfen relevant, jedoch kann man dies getrost auch in Handbüchern übernehmen. Der Aufwand, beim PDF-Output den Zugang zum Bildschirm in die normale Handlungsanleitung als Step zu integrieren, wäre zu hoch. Denn meistens wird man diese immerwiederkehrende Arbeitsschritte per conref-Mechanismus zentral pflegen und einfügen.

Vorschlag einer Zugangsauszeichnung

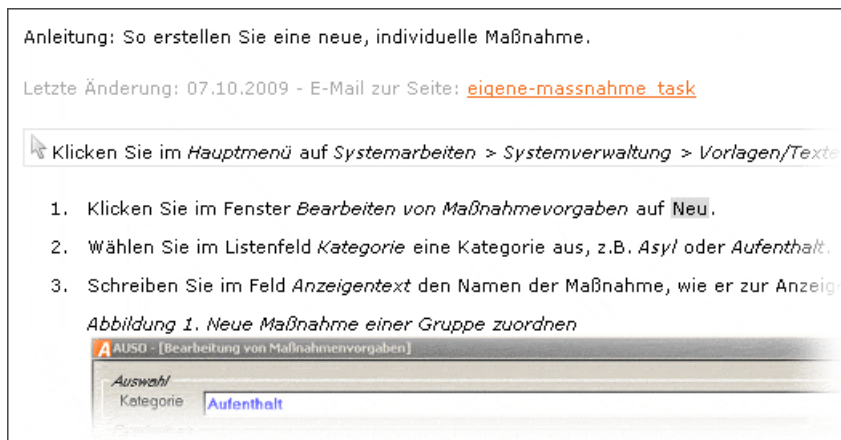


Abb. 13: Zugangsbeschreibung zum beschriebenen Bildschirm auskoppeln

Nutzer, die durch die kontextsensitive Hilfe zum Hilfetopic geleitet wurden, können gleich mit Schritt 1 anfangen.

Zugangsweg im PDF-Output darstellen

Anleitung: So koppeln Sie den Zugangspfad zu einem Bildschirm aus den Handlungsschritten einer Handlungsanleitung aus und ändern entsprechend das XSL-Stylesheet.

1. Der Hinweis auf den Zugangsweg zum Bildschirm soll mit einem Cursor-Icon und dem Wort *Zugang*: gekennzeichnet werden.

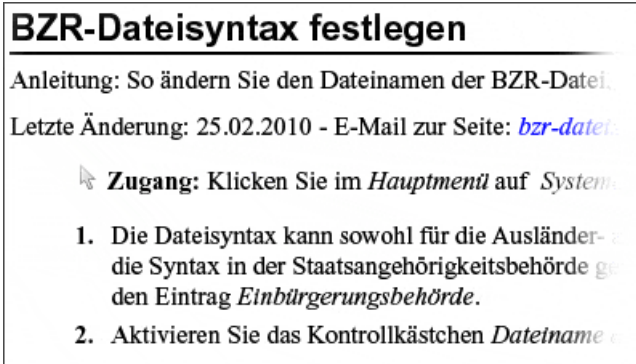


Abb. 14: Cursor-Icon vorm Zugangspfad

2. Kopieren Sie das Bild, mit welchem Sie den Zugangspfad "flaggen" möchten, in den Idiom-FO-Plugin-Ordner `C:\DITA-OT1.4.2.1\demo\fo\Customization\common\artwork`
3. Fügen Sie in der Datei `custom.xsl` im Ordner `\demo\fo\Customization\fo\attr\custom.xsl` folgendes Attributset hinzu:

```
<xsl:attribute-set name="p.zugang">
<xsl:attribute name="font-weight">bold</xsl:attribute>
</xsl:attribute-set>
```

4. Fügen Sie in der Datei `custom.xsl` im Ordner `\demo\fo\Customization\fo\xsl` folgenden Eintrag für ein Template hinzu:

```
<xsl:template match="*[contains(@class, ' topic/p ')]">
<xsl:choose>
<xsl:when test="@otherprops='zugang'">
<fo:block space-before="0.5em">
<fo:inline><fo:external-graphic src="url({concat($artworkPrefix,
'/Customization/OpenTopic/common/artwork/icon_cursor.gif')})"/></fo:inline>
<fo:inline xsl:use-attribute-sets="p.zugang"><xsl:text> Zugang:
</xsl:text></fo:inline>
<xsl:apply-templates/>
</fo:block>
</xsl:when>
<xsl:otherwise>
<fo:block xsl:use-attribute-sets="p" id="{@id}">
<xsl:apply-templates/>
</fo:block>
</xsl:otherwise>
</xsl:choose>
</xsl:template>
```

Neu ist hier die `when`-Anweisung. Die `otherwise`-Anweisung ist das normale Template für einen Absatz.

5. Versehen Sie den Absatz, der den Zugangsweg beinhalten soll, mit dem Attribut `otherprops="zugang"`. Der obige Screenshot wurde über einen Absatz im `prereq`-Tag realisiert:

```
<prereq>
<p id="konfigbzs" otherprops="zugang" outputclass="zugang">Klicken Sie
im <wintitle>Hauptmenü</wintitle> auf
<menucascade>
<uicontrol>Systemarbeiten</uicontrol>
<uicontrol>Konfiguration Module</uicontrol>
<uicontrol>BZR.</uicontrol>
</menucascade></p>
</prereq>
```

6. Optional: Weitere Überlegung sollte sein, diesen Zugangsweg nur einmal zu editieren und dann per `conref`-Mechanismus in die jeweilige Handlungsanleitung zu ziehen. Falls sich doch mal ein Bildschirm oder Menü

ändert, brauchen Sie nur in der Originaldatei den Pfad oder ähnliches ändern und durch Content Referenzierung wird die Änderung in sämtlichen Handlungsanleitungen übernommen.

7. Optional: Falls Sie wie im Beispiel um den Zugangsweg einen Rahmen ziehen möchten, vergeben Sie neben dem `otherprops`-Attribut noch ein `outputclass`-Attribut und erstellen in der CSS-Datei eine Deklaration gleichen Names.

Weitere Tipps und Werkzeuge

Installation des RenderX-Formatierers

Anleitung: So installieren Sie den RenderX-Formater für das FO-Plugin und den PDF2-Output.

1. Installieren Sie das DITA-OpenToolkit 1.5.2
2. Downloaden Sie die RenderX Personal Edition unter www.renderx.com/download/personal.htm

Personal Edition
RenderX is pleased to offer a Free, Personal Edition of our software for non-commercial use.

Free Personal Edition
Whether you are a standards developer, an open source developer, a blog administrator or just someone who would like to experiment with XSL FO (XSLFO), the Personal Edition is a great choice.

The Personal Edition is **free** and has all the functionality of the Desktop Edition **including PDF and Postscript output**. There is no qualification process, just fill in the form below and we will e-mail you a license key and instructions for download.

A valid email address is required to receive the license key.

Personal License Download

Name:

Organization:

Email:

Get your license!

The RenderX XEP Personal Edition is for non-commercial applications. PDF and Postscript files generated from the application can be distributed, posted to the

Abb. 15: Persönliche Lizenz anfordern

3. Nach Klick auf **Submit** erhalten Sie an Ihre E-Mailadresse einen Lizenzdatei vom Absender sales@renderx.com. Betreff: Download Instructions: RenderX XEP Trial Software: IhrName etc. Im Anhang befindet sich die Lizenzschlüsseldatei `license.xml`
4. Speichern Sie die Lizenzschlüsseldatei lokal ab.
5. Gehen Sie auf die Internetseite www.renderx.com/download/updates.html
6. Geben Sie den Pfad zur Lizenzdatei ein bzw. suchen Sie diese. Klicken Sie auf **Update**.

Update Current Software

Select your license key*:

C:\Dokumentation\ditalo

Now get the latest updates:

*Your license key was emailed to you when you purchased the software. It is the "license.xml" file (normally found in the XEP installation directory).

Abb. 16: Pfad der license.xml

7. Speichern die Datei `xep-Versionnummer-personal.zip` lokal ab und entpacken Sie diese.

8. Eine von den drei Dateien ist das JAR-Archiv `setup-Versionsnummer-personal.jar`. Öffnen Sie im entsprechenden Verzeichnis eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
java -jar setup-4.18-20100322-personal.jar
```
9. Scrollen Sie bis ans Endes der Lizenzvereinbarung, so dass die Schaltfläche **Yes, I Accept** aktiv wird. Klicken Sie auf die nun aktive Schaltfläche.

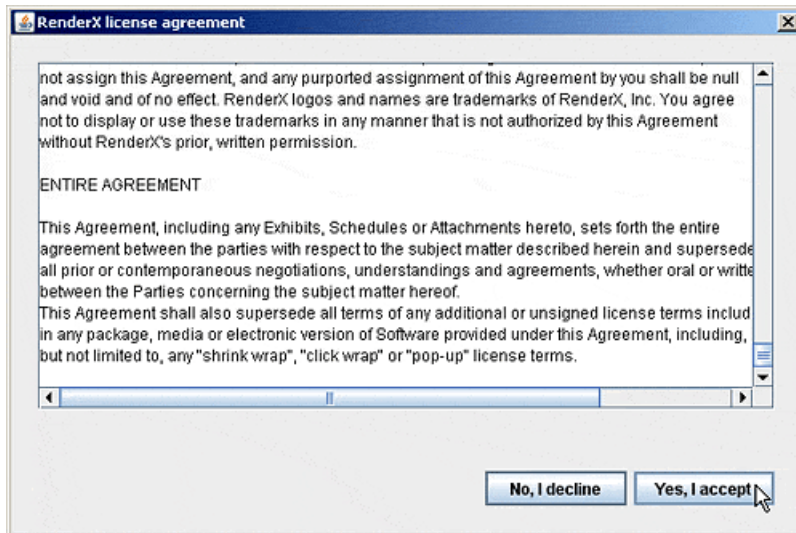


Abb. 17: Lizenzvereinbarung zustimmen

10. Wählen Sie das Verzeichnis `C:\DITA-OT1.5.2\demo\fo\lib\xep` als das Installationsverzeichnis. Dies erspart später Pfadkorrekturen im FO-Plugin.

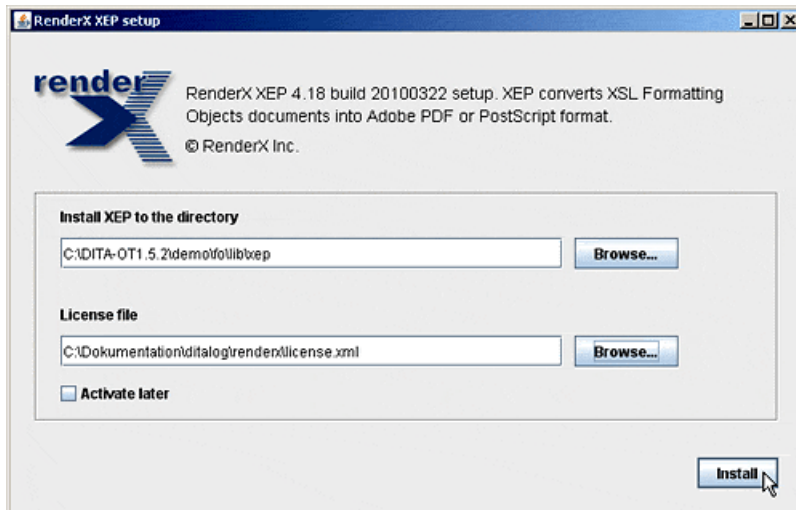


Abb. 18: Installationsverzeichnis und Lizenzschlüssel bestimmen

11. Suchen Sie die Lizenzdatei und klicken Sie auf **Install** und abschließend auf **Finish**.
12. Laden Sie die Javabibliothek `icu4j-Versionsnummer.jar` unter folgender URL herunter:
site.icu-project.org/download

ICU - International Components for Unicode

Navigation

About ICU

- [ICU Home](#)
- [Download ICU](#)

Demos & Tools

- [ICU4C Demos](#)
- [ICU4J Demos](#)
- [Data Customizer](#)

Documents

- [User Guide](#)
- [ICU FAQ](#)
- [ICU4J FAQ](#)
- [Docs & Papers](#)

API References

- [Official Release](#)
- [ICU4C \(4.4.1\)](#)
- [ICU4J \(4.4.1\)](#)

Downloading ICU

Official Releases

If you want to use ICU (as opposed to developing it), it is recommended to use the official releases rather than day-to-day development builds of the system, and they are available for download from the ICU website.

Release	ICU4C	ICU4J	Major Changes
ICU 4.4	4.4.1	4.4.1.1	CLDR 1.8, Unicode 5.2, con...
ICU 4.2	4.2.1	4.2.1.1	CLDR 1.7 update, Number S...
ICU 4.0	4.0.1	4.0.1.1	CLDR 1.6 update, Unicode 5...
ICU 3.8	3.8.1	3.8.1	CLDR 1.5 update, additional...
ICU 3.6	3.6	3.6.1	Unicode 5.0 update, UCA 5.0...
ICU 3.4	3.4.1	3.4.5	Unicode 4.1 update, CLDR 1...

Abb. 19: Download icu4j.jar

ICU4J Download

Release

[ICU4J 4.4.1.1](#) (2010-06-23)

ICU4J Download

In addition to these .jar files, the code is also tagged in [SubVersion](#) with [release-4.4.1.1](#).

File	Size	Description
icu4j-4_4_1_1-docs.jar	1.7 MB	Jar file containing the documentation.
icu4j-4_4_1_1-src.jar	14.1 MB	Jar file containing the core Java source files.
icu4j-4_4_1_1.jar	6.2 MB	Jar file containing core binaries.
icu4j-4_4_1_1.md5 📄	288 B	MD5 hash
icu4j-charsets-4_4_1_1.jar	2.2 MB	Jar file containing charset converters.
icu4j-localespi-4_4_1_1.jar	48 KB	Jar file containing locale SPI implementations.

Abb. 20: Download icu4j.jar 2. Fenster

13. Kopieren Sie die Datei in das Verzeichnis `C:\DITA-OT1.5.2\demo\fo\lib` und nennen Sie die Datei um in `icu4j.jar`.

14. Öffnen Sie die Datei `C:\DITA-OT1.5.2\demo\fo\build.xml` und ändern Sie den Wert für den `pdf.formatter` von `fop` in `xep`:

```
<property name="pdf.formatter" value="xep" />
```

15. Ändern Sie in der selben build-Datei den Eintrag für die deutsche Sprache.

```
<condition property="document.locale" value="de_DE">
```

16. Öffnen Sie über `startcmd.bat` die DITA-OT-Eingabeaufforderung und geben Sie folgenden ANT-Befehl ein:

```
ant -f integrator.xml
```

Umzug des RenderX-Formatierers

Anleitung: So kopieren Sie nach einem Update des DITA-OT den RenderX-Formatter für das FO-Plugin in das neue OT-Verzeichnis.

Voraussetzung: Sie müssen bereits für eine alte DITA-OT-Installation eine RenderX Personal Edition installiert haben.

1. Kopieren Sie den alten Customization-Ordner in das Verzeichnis `DITA-OT1.5.3\demo\fo`.
2. Kopieren Sie den alten `xep`-Ordner `DITA-OT1.5.3\demo\fo\lib\xep` und die Datei `icu4j.jar` in den `lib`-Ordner des neuen DITA-OTs.
3. Passen Sie in den 3 batch-Dateien `xep.bat`, `x4u.bat` und `validate.bat` die Pfade zum neuen DITA-OT an.
4. Ändern Sie in den Dateien `\demo\fo\build.xml` und `build_template.xml` den Wert `pdf.formatter` von `fop` in `xep`

```
<property name="pdf.formatter" value="xep" />
```

5. Ändern Sie in den selben build-Dateien den Eintrag für die deutsche Sprache.

```
<condition property="document.locale" value="de_DE">
```

6. Öffnen Sie über `startcmd.bat` die DITA-OT-Eingabeaufforderung und geben Sie folgenden ANT-Befehl ein:

```
ant -f integrator.xml
```

Leere Seiten aus der PDF-Datei entfernen.

Leere oder nicht benötigte Seiten sollen aus der PDF-Datei entfernt werden.

Problem der nicht benötigten Seiten

Als PDF-Ergebnis erhält man, auch ohne eine bookmap zu nutzen, eine PDF-Datei mit Deckblatt und Inhaltsverzeichnis. Möchte man jedoch nur eine A4-Seite für eine kleine Änderungsliste oder ähnliches erstellen, wirkt ein Deckblatt und ein Inhaltsverzeichnis mit nur einem Eintrag deplatziert.

Desweiteren entstehen oftmals leere Seiten, die nur im beidseitigem Druck einen Sinn ergeben. Der PDF-Ausdruck geschieht jedoch meistens für den einseitigen Ausdruck, worin die leeren Seiten als störend empfunden werden.

Wenn man also nicht gleich in den XSL-FO-Stylesheets Änderungen vornehmen möchte, gibt mehrere Gründe, Seiten aus der PDF-Datei zu entfernen.

Links in PDF-Datei

Bisher habe ich mich damit begnügt, die PDF-Datei aufzurufen und nur die Seiten, die ich benötigte, über einen PDF-Druckertreiber *auszudrucken*. Jedoch verschwand damit die Internet-Linkfähigkeit:



Abb. 21: Link funktioniert auch im PDF-Output

Tool PDFsam

Das kleine Tool PDFsam bewältigt zwei Probleme: Seiten entfernen und Internet-Linkfähigkeit der PDF-Datei erhalten.



Abb. 22: Teilen und wieder zusammenführen

Das Prinzip des Programm ist es, Dateien zu zerteilen, die leeren und unbrauchbaren Seiten zu löschen und nur die beschriebenen, wichtigen Seiten wieder zu einer PDF-Datei zusammenzuführen. Im Beispiel wird nach Seite 2 getrennt, weil die ersten beiden Seiten das nicht benötigte Deckblatt und das Inhaltsverzeichnis darstellen. Und es wird nach Seite 7 getrennt, weil Seite 8 das Stichwortverzeichnis ist, welches für eine kurze Installationsanleitung nicht vonnöten ist.

Im obigen Beispiel kommen wir also mit Teilen aus. Sind zwischendurch Leerseiten zu entfernen, müssen die einzelnen wichtigen Zwischenstücke am Ende wieder zu einer PDF-Datei zusammengeführt werden.

Index

A

A4 3
 Acrobat Reader 4
 attribute-set 7, 8

B

basic-settings.xml 3
 bookmap 10
 planen 10

C

custom.xsl 7, 8

D

de_DE.xml 7
 DITA Yahoo Users Group 4
 DITA-OT updaten 18
 DITA-OT1.5.2 16
 ditaval-Datei 12

E

Einzelkapitel im PDF-Inhaltsverzeichnis 5

F

Flagging im PDF-Output 13
 FO Plugin 3, 7, 8
 Fußzeile im PDF 7, 8

H

hängende erste Zeile 6

I

icu4j.jar 17
 Inhaltsverzeichnis 10
 insertBodyOddFooter 8

L

layout-master.xml 3
 Linking in PDFs 12
 Links in PDF 19

P

page-height 3
 page-width 3
 Paragraphensymbol im indexterm 9
 PDF leere Seiten entfernen 19
 PDF Split and Merge 19
 PDF-Metaangaben 4
 Produktattribut 12

R

RenderX Personal Edition 16
 rootprocessing.xsl 4

S

Seitengröße A4 3
 static-content.xsl 8

T

text-indent 6
 Texteinrückung 6
 Titel 4

V

Verfasser 4

X

XEP umziehen 18
 xref-Links 11

Z

Zugang zum Bildschirm auskoppeln 13