

Wissenschaftliche Arbeiten mit BibTeXML in DocBook XML schreiben



by Egon Willighagen
<egonw(at)linuxfocus.org>



About the author:

Hat seinen Master in Chemie an der Universität von Nijmegen gemacht und arbeitet im Rahmen seiner Doktorarbeit über molekulare Darstellung an derselben Universität. Spielt Basketball und programmiert Java-Anwendungen.

Abstract:

In diesem Artikel werde ich Dir zeigen, wie Du BibTeX als Referenzsystem mit DocBook XML benutzen kannst. Um diesen Prozeß zu erleichtern, habe ich einige Werkzeuge entwickelt, die als Pakete in der JReferences Distribution enthalten sind.

Einführung

LaTeX Benutzer wissen, wie hilfreich BibTeX ist. Es ist ein sehr komfortables Werkzeug, um Referenzen zu anderer wissenschaftlicher Literatur hinzuzufügen, ohne dass man sich viel um die tatsächliche Ausgabe Gedanken machen muss, weil die in jedem Fall richtig sein wird – zum Beispiel werden die richtigen Einstellungen vorgenommen, aber die Typen *nicht* von Hand eingestellt. Genau wie bei LaTeX selbst. Ausserdem ist es in wissenschaftlicher Literatur üblich, die Referenzen mit Exponenten-Nummern zu versehen, wie diese ¹. Und diese Nummern sollten fortlaufend vergeben sein. BibTeX kümmert sich auch hierum.

DocBook wächst Tag für Tag zu meinem Lieblings Autorentool heran, weil es eine saubere XML-basierte Syntax hat, einen hervorragenden Support für die Erstellung von Webseiten (zum Beispiel ist die CDK Website, <http://cdk.sf.net/>, komplett in DocBook geschrieben) und man pages bietet. Der nächste Schritt für mich war, DocBook für wissenschaftliche Artikel zu benutzen. Also brauchte ich BibTeX für DocBook. Daher habe ich JReferences geschrieben.

JReferences macht etwas mehr als BibTeX. Wie BibTeX hat es Werkzeuge, um die Referenzen automatisch aus einer einfachen Text-Datenbank zu numerieren, aber es kann noch mehr. Es unterstützt mehr Formate

(sowohl als Eingabe, als auch als Ausgabe) und es hat eine MySQL-Schnittstelle, auf die über eine PHP gesteuerte Oberfläche zugegriffen werden kann. Es versucht auch, eine Referenz Datenbank, wie zum Beispiel EndNote zu sein. Wie auch immer, obwohl es ein Open Source Projekt ist (und der GPL Lizenz unterliegt), hat es außer mir noch nicht sehr viele Entwickler angezogen und die Weiterentwicklung geht recht langsam. Das heißt aber nicht, dass es nicht nützlich ist und ich werde Dir genau das in diesem Artikel beweisen.

Bei Veröffentlichung dieses Artikels war die Version 0.7.2 von JReferences aktuell und dieser Artikel behandelt diese Version.

Ein DocBook Artikel

Sieh' Dir das Beispiel im JReferences Paket an.

```
<?xml version="1.0"?>
<!DOCTYPE article PUBLIC
"-//JReferences//DTD DocBook JReferences Module //EN"
    "../dtd/jreferences.dtd" []>
<article>

  <jref:mode>Number</jref:mode>

  <articleinfo>
    <title>Test Article</title>
    <author><firstname>Egon</firstname>
      <surname>Willighagen</surname></author>
    <date> 3 May 2000</date>
  </articleinfo>

  <section>
    <title>Some section</title>
    <para>This is a text with a reference
    <jref:cite id="Steinbeck99"/>.</para>
    <para>And now for some more serious tests, we
    add a second reference <jref:cite id="Bachrach99"/>.
    And again the first reference <jref:cite id="Steinbeck99"/>.
    </para>
  </section>

  <jref:reflist/>

</article>
```

Ich werde dieses Beispiel Zeile für Zeile erklären. Die erste Zeile ist Deine (optionale) Zeile mit der Kennzeichnung der Syntax dieser Datei als XML. Die zweite bis vierte Zeile sagen aus, dass die XML Sprache, die hier verwendet wird, DocBook ist, dass aber anstelle der normalen DTD das JReference Modul benutzt wird. Normales DocBook XML weiß nichts über JReferences und die Verwendung dieser DTD würde ungültige Dokumente ergeben. Das JReferences Modul andererseits kennt sowohl DocBook als auch JReferences (für DocBook Spezialisten: es kennt zur Zeit noch kein SVG oder MathML). Daher gibt Dir die Verwendung dieses Moduls wieder die Möglichkeit der Validierung Deines Dokuments. Das oben genannte Beispiel ist ein gültiges DocBook Dokument.

Die fünfte Zeile enthält das Startkennzeichen (tag) für das Artikel-Element. So weit, so gut. In der siebten Zeile beginnt der Spaß: Das erste jref Element. Das <jref:mode> Element wird benutzt, um JReferences mitzuteilen, welcher Typ von Referenznummer benutzt werden soll. In der Einführung habe ich bereits

geschrieben, dass normalerweise hochgestellte Nummern verwendet werden. Es gibt aber trotzdem viele Alternativen. JReferences unterstützt [1], ¹, und [Steinbeck99]. Das letztere ergibt den Code, der in der Referenz benutzt wird. Das Beispiel benutzt die erstgenannte Option.

Die nächsten paar Zeilen enthalten etwas DocBook Inhalt und die nächste wirklich interessante Zeile ist Zeile 17. In dieser Zeile wird die erste Referenz zitiert. LaTeX Anwender würden hierfür `\cite{ }` schreiben; die JReferences Syntax ist `<jref:cite id="IrgendeineID"/>`. Die ID stimmt mit einer Referenz in der Datenbank überein, was später noch näher erläutert wird. Der nächste Absatz des Abschnitts enthält zwei weitere Zitate, eines davon bezieht sich wiederum auf die erste Referenz.

Um die eigentlichen Referenzen einzubinden, wird in Zeile 24 `<jref:reflist />` verwendet. Dieser JReferences Befehl wird zu einer DocBook formatierten Liste von Referenzen umgewandelt, und zwar in der Reihenfolge, in der die Referenzen zitiert werden.

Die BibTeXML Datenbank

Das JReferences System benötigt eine Datenbank, also so etwas wie die *.bib-Dateien in LaTeX/BibTeX. JReferences bietet eine Unterstützung für eine BibTeXML- Datenbank, aber ebenso für andere (zum Beispiel MySQL). BibTeXML wurde von Vidar Gundersen und Zeger Hendrikse entwickelt. Das Beispiel in der JReferences Distribution (0.7.2) verwendet BibTeXML noch nicht, aber unser Beispielartikel würde folgende BibTeXML Datei haben:

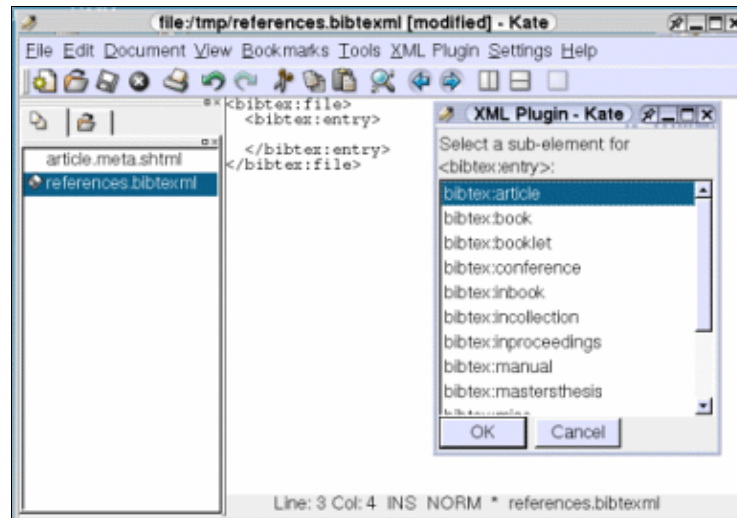
```
<?xml version="1.0" encoding="UTF-8"?>
<bibtex:file xmlns:bibtex="http://www.bitjungle.com/~bibtex/">

<bibtex:entry bibtex:id="Steinbeck99">
  <bibtex:article>
    <bibtex:title>JChemPaint - Using
      the Collaborative Forces of the Internet to
      Develop a Free Editor for 2D Chemical
      Structures</bibtex:title>
    <bibtex:author>Steinbeck, C. and
      Krause, S. and
      Willighagen, E.</bibtex:author>
    <bibtex:year>2000</bibtex:year>
    <bibtex:volume>5</bibtex:volume>
    <bibtex:pages>93-98</bibtex:pages>
  </bibtex:article>
</bibtex:entry>

<bibtex:entry bibtex:id="Bachrach99">
  <bibtex:article>
    <bibtex:title>End-User Customized Chemistry Journal
      Articles</bibtex:title>
    <bibtex:author>Bachrach, S. and
      Krassavine, A. and
      Burleigh, D.</bibtex:author>
    <bibtex:journal>J.Chem.Inf.Comput.Sci.</bibtex:journal>
    <bibtex:year>1999</bibtex:year>
    <bibtex:volume>39</bibtex:volume>
    <bibtex:pages>81-85</bibtex:pages>
  </bibtex:article>
</bibtex:entry>

</bibtex:file>
```

Die zweite Zeile enthält das Start Tag für das Hauptelement <bibtex:file>. Eine solche Datei enthält ein oder mehrere <bibtex:entry> Elemente. Und jeder Eintrag besteht aus einem BibTeX XML Referenztypem: article, book, inbook, incollection, unpublished, misc und others. Jede dieser Referenzen enthält wiederum spezifische Elemente für diesen Typ, aber eine ganze Menge von ihnen sind allgemein gültig, wie zum Beispiel <bibtex:title> und <bibtex: year>. Die JReferences Distribution enthält die BibTeX XML DTD, so dass jeder XML Editor, der mit DTD umgehen kann, auch leicht BibTeX Dokumente bearbeiten kann. Außerdem enthält JReferences Meta DTDs für Kate in KDE 3.x (siehe [DocBook XML Dokumente bearbeiten](#)) die automatisch nach \$HOME/.kde/share/apps/katexmlplugin installiert werden.



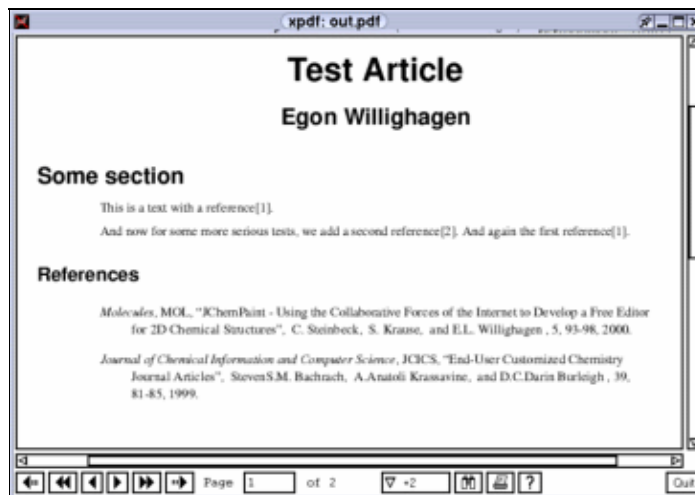
BibTeX XML Dateien mit Kate, dessen XML Plugin und JReferences' BibTeX XML Meta DTD bearbeiten.

Ein Literaturverzeichnis erstellen

Sieh' Dir die beiden Beispiele oben an. Das DocBook Dokument ist als article.docbookxml und die Referenzdatenbank ist als references.bibtexml gespeichert. JReferences enthält zur Zeit noch kein Werkzeug wie das bibtex-Programm, aber derselbe Effekt kann mit ein paar Befehlen erreicht werden. Die untenstehenden Befehle gehen davon aus, dass Du JReferences auf einem UNIX-ähnlichen System, wie zum Beispiel Linux, installiert hast:

```
jref-clear --filedb  
jref-set --filedb --bibtexml references.bibtexml  
jref-number --filedb article.docbookxml > article-numbered.docbookxml
```

Die hieraus erzeugte Datei namens article-numbered.docbookxml ist ein gültiges DocBook XML 4.1.2 Dokument ohne irgendein <jref:*> Element und kann von jedem anderen Tool benutzt werden, um DocBook XML Dokumente zu konvertieren, beispielsweise nach PDF. (Siehe zum Beispiel [PDF Dokumente erzeugen mittels DocBook](#)).



Die erzeugte PDF Datei mit nummerierten Referenzen und einem eingebundenen Literaturverzeichnis.

Das ist alles, was Du wissen musst. Das heißt...

Formatierungsstile

... es gibt noch etwas interessantes. BibTeX unterstützt Stilvorgaben, weil die meisten Zeitschriften bestimmte Anforderungen an die Formatierung des Literaturverzeichnisses stellen. JReferences enthält zur Zeit nur zwei Stile. Der erste ist das Standard DocBook XML Format, was nicht wirklich ein Stil ist. Aber es gibt auch noch einen Stil, der von der American Chemical Society (ACS) gefordert und von JReferences unterstützt wird.

Das `<jref:reflist>` Element hat ein `@style` Attribut, mit dem man den Stil, der anstelle des Defaultstils verwendet werden soll, einstellen kann. Um den ACS-Stil zu verwenden, mußt Du Zeile 23 ersetzen durch

```
<jref:reflist style="ACS"/>
```

JReferences installieren

JReferences erfordert mindestens eine Java 1.3 Installation, sowie Xerces, Log4J und die DocBook XML DTD 4.1.2. Einige Tools brauchen weitere Werkzeuge und Sprachen, wie zum Beispiel Python (für die Übersetzung von BibTeX zu BibTeXML) und Perl (um EndNotes BibTeX Ausgabe zu bereinigen).

Wenn diese Voraussetzungen gegeben sind, kann JReferences mit den folgenden Befehlen installiert werden:

```
./configure --prefix=$HOME  
make  
make install
```

Wenn einige Tools nicht gefunden werden, probiere es mit den folgenden Optionen: `--with-xercesdir`, `--with-log4javadocir` und `--with-sgmlidir`. Für nähere Informationen über diese Optionen tippe `./configure --help`.

Das Projekt

JReferences ist nun rund zwei Jahre alt und obwohl es sehr oft heruntergeladen worden ist, habe ich nicht viel Rückmeldungen bekommen. Abgesehen mal von meinen eigenen Erfahrungen. In den letzten paar Monaten wurde JReferences erfolgreich genutzt, um einen wirklich wissenschaftlichen Artikel zu schreiben. Wie auch immer, wie bei jedem guten Open Source Projekt ist jeder Kommentar, Bug Report, Patch, jede Idee und Erfolgsgeschichte auf der JReferences [Projekt Seite](#) willkommen.

Referenzen

- [DocBook XML](http://docbook.sf.net), <http://docbook.sf.net>
- [BibTeX XML](http://bibtexml.sf.net), <http://bibtexml.sf.net>
- [JReferences](http://sf.net/projects/jreferences), <http://sf.net/projects/jreferences>

<p>Webpages maintained by the LinuxFocus Editor team © Egon Willighagen "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org</p>	<p>Translation information: en --> -- : Egon Willighagen <egonw(at)linuxfocus.org> en --> de: Michael Weinrich <m.weinrich(at)web.de></p>
---	---

2005-01-11, generated by lfparsr_pdf version 2.51