

# DITA und das Open Toolkit in der Praxis

by Ferdinand Soethe

*Single Source Publishing mit DITA und dem Open Toolkit entwickelt sich immer mehr zu einer echten Alternative für die technische Dokumentation. Allerdings erschwert die recht technisch gehaltene Dokumentation von Installation und Erstkonfiguration den Einstieg. Der Vortrag möchte anhand eines umfangreichen Handbuch-Projekts Wege aufzeigen, DITA schnell und einfach für die technische Dokumentation nutzbar zu machen.*

## Table of contents

1	Einleitung.....	3
1.1	Zu meiner Person .....	3
1.2	Themen heute.....	3
1.3	Unterlagen im Internet.....	3
2	Woher kommt DITA und wo will es hin?.....	3
2.1	Was ist DITA?.....	3
2.2	Woher kommt DITA?.....	3
2.3	DITA und OASIS.....	3
3	Wie funktioniert DITA?.....	4
3.1	Kernaspekte.....	4
3.2	Themenzentrierung.....	4
3.3	Flexible Publikationsstruktur.....	5
3.4	Modulare Wiederverwendung.....	7
3.5	Erweiterbarkeit.....	7

3.6 Bedingte Elemente.....	8
4 Das DITA-Open Toolkit.....	8
4.1 Was ist das DITA-OT?.....	8
4.2 Was leistet DITA-OT.....	8
4.3 DITA-Praxis mit dem Open Toolkit.....	8
4.4 Einrichtung eines Handbuch-Projekts.....	9
5 DITA-Features für Fortgeschrittene.....	10
5.1 DITA Targets.....	10
5.2 DITA für Fachdomänen.....	10
6 Tips für Umsteiger.....	12
6.1 Nützliche Tools für DITA-Projekte.....	12
6.2 Quellen.....	12

## 1 Einleitung

### 1.1 Zu meiner Person ...

Ferdinand Soethe

- Softwarearchitekt, Autor und Berater
- Mitglied des Apache Forrest PMC
- Arbeitsschwerpunkt: SSP-Systeme

### 1.2 Themen heute

- DITA
- Das DITA-OT
- DITA Praxis
- Nützliche Tools

### 1.3 Unterlagen im Internet

Die Unterlagen zu diesem Vortrag finden Sie im Internet unter

<http://soethe.net/Vortraege/Dita>

Sie haben dort auch die Möglichkeit, Ihrer Meinung zu diesem Vortrag Ausdruck zu verleihen.

## 2 Woher kommt DITA und wo will es hin?

### 2.1 Was ist DITA?

Architektur für Single Source Publishing

- Regeln für Informationsaufbereitung
- Baukasten von Grammatiken und Elementen
- Erweiterbares Verarbeitungskonzept

### 2.2 Woher kommt DITA?

- Produkt der IBM
- Nachfolger für eigene SGML-Dokumentationssysteme

### 2.3 DITA und OASIS

- DITA wird 2004 von IBM an OASIS gespendet

- Seit 2005 ist DITA 1.0 OASIS Standard
- August 2007: OASIS DITA Version 1.1.

### 3 Wie funktioniert DITA?

#### 3.1 Kernaspekte

- Themenzentrierung
- Flexible Publikationsstruktur
- Modulare Wiederverwendung
- Erweiterbarkeit

#### 3.2 Themenzentrierung

##### 3.2.1 Themenzentrierung

DITAs größte Informationsmodule (Topics) sind

- in sich abgeschlossen
- typisiert

##### 3.2.2 DITA Topic-Ideal

Das ideale DITA-Topic ist

- so groß wie nötig
- so klein wie möglich
- frei von Abhängigkeiten
- überall einsetzbar

##### 3.2.3 DITA Topics

Typisierung von Topics als Information-Types zwingt Autoren zur Klarheit ihrer Aussagen.

- CONCEPT
- TASK
- REFERENCE

##### 3.2.4 CONCEPT

- beantwortet "Was-ist-Fragen"
- Hintergrund-Informationen
- oftmals eine erweiterte Definition
- einfach strukturiert

### 3.2.5 TASK

- beantwortet Wie-kann-ich-Fragen
- beschreibt Vorgehen konkret oder abstrakt
- enthält Element für Voraussetzungen, Schritte, Ergebnisse
- verknüpft zu weiterführenden Aktionen

### 3.2.6 REFERENCE

- dokumentiert Fakten
- domänenspezifische Struktur

### 3.2.7 Shell-Topics

Topics sind innerhalb eines Shell-Topics

- beliebig rekombinierbar
- schachtelbar

## 3.3 Flexible Publikationsstruktur

### 3.3.1 Strukturierung mit DITA MAP

MAPS kombinieren Topics zu Publikationen.

- Einfache Listen
- Komplexe Hierarchien
- Modulare Wiederverwendung von MAPS

### 3.3.2 Beziehungsdefinition

Attribute in den Elementen der Map definieren

- Art der Zusammengehörigkeit  
Familie, Sequenz, Hierarchie
- Wichtigkeit der Beziehung  
`importance="required"`

### 3.3.3 Weitere Beziehungen

Beziehungen, die sich nicht aus der Gliederung ergeben können durch zusätzliche Abschnitte innerhalb der map beschrieben werden

- Related-Links

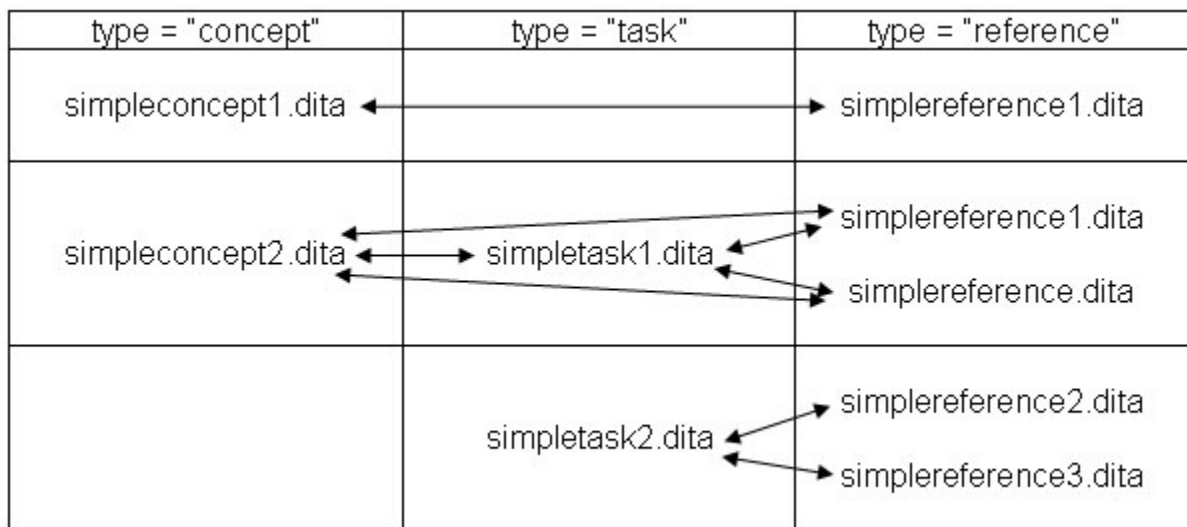
### 3.3.4 Related-Links

Mit einem related-links-element können Beziehungen zwischen Topics gezieht erklärt und beschrieben werden.

```
<related-links> <link href="audience.dita" scope="peer"/> <link href="../evaluating/
framework.dita" scope="local"/> </related-links>
```

### 3.3.5 Reltable

Relationship-Tables sind eine sehr effiziente Form, Beziehungen zu definieren.



Quelle: Kylene Bruski, Comtech Services, Inc.

### 3.3.6 Reltable Source

Im Source-Code sind die allerdings sehr schwer verständlich:

```
<reltable> <relheader> <relcolspec type="concept"/> <relcolspec type="task"/>
<relcolspec type="reference"/> </relheader> <relrow> <relcell> <topicref href="concepts/
about_produce.dita"/> </relcell> <relcell> <topicref href="tasks/choosing_produce.dita"/>
</relcell> <relcell> <topicref href="reference/produce.dita"/> </relcell> </relrow> <relrow>
<relcell> <topicref href="concepts/about_cannedgoods.dita"/> </relcell> <relcell> <topicref
href="tasks/buying_cannedgoods.dita"/> </relcell> <relcell> <topicref href="reference/
cannedgoods.dita"/> </relcell> </relrow> </reltable>
```

### 3.3.7 Praktische Konsequenz

- MAPS als wichtigen Aspekt der Wiederverwendung nutzen lernen

- Ort der Beziehungsdefinition genau überlegen

### 3.3.8 Weiterführende Informationen

- Tutorial "DITA Linking and relationship tables"  
[http://www.ditausers.org/tutorials/presentations/Stark\\_Links/](http://www.ditausers.org/tutorials/presentations/Stark_Links/)
- Linking DITA Topics Through Relationship Tables  
<http://www.infomanagementcenter.com/newsletter/200511/second.htm>

## 3.4 Modulare Wiederverwendung

### 3.4.1 Wiederverwendung mit conref

Content-References (conref) verbessern die Möglichkeiten von x-include und SGML conref.

### 3.4.2 Conref-Praxis

Ein komplettes Querverweiselement einfügen:

`<p>Mit der Maske <xref conref="bde_masken#topic-1/bde_maske_abrechnung_selbstzahler"/> bereiten Sie die Abrechnung für Selbstzahler vor und führen sie durch.</p>`

Das einzufügende Element muss lediglich eine eindeutige ID innerhalb seiner Datei besitzen.

```
<ul> <li> <xref href="bde_maske_abrechnung_selbstzahler.xml"
id="bde_maske_abrechnung_selbstzahler"> <uicontrol><wintitle>Abrechnung
Selbstzahler</wintitle></uicontrol> </xref>
```

### 3.4.3 Praktische Konsequenzen

- Verwendung von Conref erleichtert Anpassungen
- Conref ermutigt zu strukturierter Arbeitsweise

## 3.5 Erweiterbarkeit

Architektur und Verarbeitung zielen auf leichte Erweiterbarkeit in mehrere Richtungen

- Re-kombination von Info-Types zu komplexeren Strukturen
- Schaffung neuer Information-Types
- Schaffung neuer Elementarer Bausteine

### 3.6 Bedingte Elemente

Von Haus aus kann jedes DITA-Element mit Attributen versehen werden, die eine spätere Filterung nach

- Produkt
- Betriebssystem
- Zielgruppe
- weiteren frei definierbaren Eigenschaften

ermöglichen.

## 4 Das DITA-Open Toolkit

### 4.1 Was ist das DITA-OT?

- Eine Open Source Lösung für DITA
- Referenzimplementierung der DITA-Architektur
- Motor in vielen kommerziellen DITA-Lösungen

### 4.2 Was leistet DITA-OT

- Allgemeine DITA-Verarbeitung (conref-Auflösung)
- Generische Implementierung verschiedener Transformations-Targets  
XHTML, PDF, HTML-Help ...
- Default-Verarbeitung für Spezialisierungen
- Anpassbare Batchlösungen für die automatisierte Verarbeitung

### 4.3 DITA-Praxis mit dem Open Toolkit

#### 4.3.1 Systemvoraussetzungen

- Java-fähiges Betriebssystem  
Windows, Linux, OS/X
- Java

#### 4.3.2 Java installieren

- Java herunterladen  
<http://www.java.com/de/download/index.jsp>
- Installer ausführen

### 4.3.3 DITA OT herunterladen

- Das DITA Open Toolkit finden Sie im Internet unter <http://dita-ot.sourceforge.net/>
- Version 1.4.x.x
- full-easy-install
- Wahlweise ZIP oder tarbar

### 4.3.4 Installationspaket auspacken

- Betriebssystemabhängig  
Windows: Unzippen
- Inhalt komplett in ein Verzeichnis mit Unterverzeichnissen auspacken.
- Sinnvollerweise in einem Programmverzeichnisbereich

Windows: C:\Programme

Die Datei EvaluateOT.html beschreibt auf Englisch ausführlich die weitere Installation.

### 4.3.5 DITA Demo testen

- startcmd.bat ausführen
- ant samples.web -f build\_demo.xml ausführen
- ant all -f build\_demo.xml

## 4.4 Einrichtung eines Handbuch-Projekts

### 4.4.1 DITA Projektverzeichnis

Jedes DITA-Projekt sollte in einem separaten Verzeichnis leben.

- Leeres Verzeichnis
- Sinnvollerweise NICHT im DITA-Verzeichnis

### 4.4.2 ANT-Projektdatei

- Kopie von ant/sample\_xhtml.xml ins Projektverzeichnis
- Anpassungen erforderlich

### 4.4.3 Batchdatei

- Vereinfacht den Aufruf
- Beschleunigt die Arbeit

- Gibt Anfängern Sicherheit

## 5 DITA-Features für Fortgeschrittene

### 5.1 DITA Targets

In der Praxis interessant ist die Unterstützung vieler verschiedener Zielformate:

- XHTML
- Windows HTML Hilfe
- Unix-Hilfe
- Java-Help
- Eclipsehelp
- PDF
- RTF

### 5.2 DITA für Fachdomänen

#### 5.2.1 DITA für Fachdomänen

Die Anpassung von DITA an neue Erfordernisse kann auf verschiedenen Ebenen erfolgen.

#### 5.2.2 Kombinierbarkeit der Informationsmodule

In sich geschlossene Architektur der Topic-Verarbeitung

#### 5.2.3 Configuration

Configuration beschreibt die Entstehung neuer Topic-Typen als Hülle um vorhandene Topic-Typen

- Neue Topic-Typen durch Kombination vorhandener Topic-Typen
- Formal deklariert

#### 5.2.4 Spezialisierung (Specialisation)

- Neue Topic-Typen (Structural Specialisation)
- Neue Elemente oder Attribute (Domain Specialisation)

#### 5.2.5 Domains

DITA selbst gibt Beispiele für Domain-Spezialisierungen  
software

msgph, msgblock, msgnum, cmdname, varname, filepath, userinput, systemoutput

programming

codeph, codeblock, option, var, parmname, synph, oper, delim, sep, apiname, parml, plentry, pt, pd, syntaxdiagram, synblk, groupseq, groupchoice, groupcomp, fragment, fragref, synnote, synnoteref, repsep, kwd

user interface

uicontrol, wintitle, menucascade, shortcut

### 5.2.6 Wann/wo muss ich DITA erweitern

- Zusätzliche Transformationsziele
- Branding
- Custom Styling

### 5.2.7 Wie sollte DITA erweitert werden?

Plugins sind der sinnvollste Weg Erweiterungen vorzunehmen

- mit einfachen Mitteln zu realisieren
- leichter auf neue Versionen zu aktualisieren
- leichter weiterzugeben

### 5.2.8 Einfache Erweiterungen

Einhängepunkte (Hooks) im System vereinfachen viele Erweiterungen auf wenige Zeilen Code:

```
<?xml version="1.0" ?>
- <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
- <xsl:template name="gen-user-header">
  - <div>
    
  </div>
</xsl:template>
</xsl:stylesheet>

<?xml version="1.0" encoding="UTF-8" ?>
- <plugin id="org.dita.brandheader">
  <feature extension="dita.xsl.xhtml" value="xsl/header.xsl" type="file" />
</plugin>
```

## 6 Tips für Umsteiger

### 6.1 Nützliche Tools für DITA-Projekte

#### 6.1.1 XML-Editoren

- Das Eclipse-Webtools-Paket bietet eine sehr brauchbare Open Source-Lösung mit grammatikfähigem Editor und Revisionsverwaltung als Plugin.
- oXygen  
<http://www.oxygenxml.com/>  
ist eine hochwertige Lösung für Programmierung und Authoring (aber kommerziell!).
- XMLMind  
<http://www.xmlmind.com/xmleditor/>
- Interessant ist auch der Online-Editor unter  
<http://www.ditastorm.com/>

#### 6.1.2 Revisions-Verwaltung

DITA selbst bietet keine Revisionsverwaltung.

Sie kann aber mit den Open Source-Werkzeugen *Subversion* (<http://subversion.tigris.org/>) und *Tortoise* (<http://tortoisesvn.tigris.org/>) leicht nachgerüstet werden.

### 6.2 Quellen

- Kostenlose Mitgliedschaft, Zugriff auf Tools  
<http://www.ditauers.org/>
- Online DITA Referenz  
<http://www.ditainfocenter.com/>
- Spezialisierungs-Tutorial  
<http://www.xiruss.org/tutorials/dita-specialization/>
- <http://dita.xml.org/http://www.infomanagementcenter.com/newsletter/200511/second.htm>