



# **Verwendung von nicht- westlichen Zeichensätzen in den Firebird-Dokumentationen**

Eine Anleitung zum Einbetten von Schriftarten für Firebird Dokumentenersteller und Übersetzer

Paul Vinkenoog

20. Juli 2013 – Version 1.3.1-de - deutsche Version

Übersetzung ins Deutsche: Martin Köditz

---

---

## Inhaltsverzeichnis

Einleitung .....	3
Wie werden PDF-Dateien erstellt? .....	3
Schritt 1: Finden der benötigten Schriftarten .....	4
Schritt 2: Überschreiben der Stylesheet-Schriftarten .....	5
Schritt 3: Erstellung der Metric-Dateien .....	6
Schriftarten-Sammlungen .....	7
Schritt 4: Erstellung einer FOP-Benutzerkonfigurationsdatei .....	7
Schritt 5: Erstellung des PDF und Ausführung des Commits .....	9
Nachwort .....	9
Anhang A: Dokumenthistorie .....	11
Anhang B: Lizenzhinweis .....	12

---

## Einleitung

Wenn Sie nicht-westliche Zeichensätze in Ihrer Firebird-Dokumentation verwenden, wird die HTML-Version wahrscheinlich gut aussehen. Moderne Browser werden kaum Probleme mit diversen Sprachen und Schriften haben, solange die notwendigen Schriftarten auf dem System vorhanden sind. Möchten Sie nun aber die PDF-Version erstellen, müssen einige Punkte beachtet werden:

1. Finden Sie die Schriften, die für body-Texte, Titel und Monospaced-Schriften benötigt werden.
2. Überschreiben Sie die Standard-Fonts, die in den Stylesheets angegeben sind.
3. Erstellen Sie *Metric-Dateien* für die Schriften, die Sie verwenden wollen.
4. Erstellen Sie eine *FOP-Benutzerdatei* mit den Anleitungen zum Einbetten der Schriften.
5. Erstellen Sie die PDF. Wenn alles so läuft wie es soll, führen Sie einen Commit der Schritte 2 bis 4 in das CVS durch.

Wenn Sie die erste Person sind, die eine Dokumentation in einer Sprache erstellt, müssen Sie alle o.a. Schritte durchführen. Ist dies nicht der Fall, können Sie ein existierendes Setup verwenden. Möglicherweise müssen Sie die FOP-Benutzerkonfigurationsdatei bearbeiten, da die Orte der Schriftdateien von System zu System unterschiedlich sein können. Siehe auch „[Wichtig](#)“-Kasten Schritt 4.

Es wird vorausgesetzt, dass Sie bereits mit der Arbeit mit DocBook XML-Dateien in Ihrer eigenen Sprache vertraut sind. Speichern Sie die Dateien in einer Unicode-Codierung, beispielsweise UTF-8. XMLMind, SciTE, Windows Notepad und andere Editoren haben keine Probleme hiermit (XMLMind speichert standardmäßig in UTF-8). ConText hingegen kann Unicode nur als UTF-16 speichern, aber soweit bekannt, stellt auch das kein Problem für die Build-Tools dar.

## Wie werden PDF-Dateien erstellt?

Um die auszuführenden Schritte besser zu verstehen, folgt nun ein kurzer Überblick, wie Ihr DocBook-Quelltext zu PDF konvertiert wird.

### *Schritt 1: DocBook -> XSL-FO*

Ein sogenannter *XSL transformer* mit dem Namen Saxon liest den DocBook XML-Quelltext, wandelt ihn ins *XSL-FO*-Format und speichert ihn im Verzeichnisbaum unter `manual/inter/fo`. Das FO in XSL-FO steht für *Formatting Objects*. Wie DocBook selbst, ist dies ein XML-Format, jedoch anzeigeorientiert. Nachstehend ein typischer Auszug von XSL-FO:

```
<fo:block keep-together="always" margin-left="0pc"
  font-family="sans-serif,Symbol,ZapfDingbats">
  <fo:block font-family="sans-serif" font-weight="bold"
    keep-with-next.within-column="always"
    space-before.minimum="0.8em" space-before.optimum="1.0em"
    space-before.maximum="1.2em"
    color="darkblue" text-align="start">
```

```
<fo:block font-size="19.8pt">SQL Syntax</fo:block>
</fo:block>
</fo:block>
```

Das Saxon-Tool finden Sie unter `manual/lib/saxon.jar`. Um verstehen zu können *wie* wir die DocBook-Teile zu XSL-FO konvertieren möchten, lädt Saxon ebenfalls die Transformations-Stylesheets. Die Standard-DocBook-Stylesheets befinden sich in `manual/src/docs/docbook`, die eigenen Anpassungen in `manual/src/docs/xsl`.

In diesem ersten Schritt wird die DocBook -> XSL-FO Transformation vom `fo build`-Target ausgeführt. Wenn Sie den Befehl „**build pdf...**“ eingeben, wird das `fo`-Ziel bereits intern aufgerufen, Sie können es jedoch auch explizit starten, um nur die XSL-FO-Dateien zu bauen, ohne die Erstellung des PDFs.

### Schritt 2: XSL-FO -> PDF

Die XSL-FO-Datei wird zum PDF konvertiert. Dies geschieht durch das Werkzeug Apache FOP (*Formatting Objects Processor*). Das Ergebnis wird im Verzeichnisbaum unter `manual/dist/pdf` gespeichert. Weder Stylesheets noch die originale DocBook-Quelle sind hiervon betroffen. Apache FOP ist zu finden in `manual/lib/fop.jar`.

Dieser Schritt wird ausgeführt mittels des `fo2pdf build target`. Auch dieses wird intern vom `pdf target` aufgerufen. Aber auch hier können Sie die `.fo`-Dateien selbst bearbeiten (was manchmal notwendig ist).

Die folgenden Abschnitte, werden durch die in der Einleitung beschriebenen Schritte führen. Die Tabelle zeigt, welcher der jeweiligen Schritte den DocBook-to-PDF Erstellungsprozess beeinflusst:

Schritt	Beschreibung	Beeinflussungen
1	Finden der benötigten Schriftarten	(vorbereitender Schritt)
2	Überschreiben Stylesheet-Fonts	Schritt 1: XSL-FO-Erstellung
3	Erstellung der Metric-Dateien	Schritt 2: PDF-Erstellung
4	Erstellung der FOP-Benutzerkonfigurationsdatei	Schritt 2: PDF-Erstellung
5	PDF "bauen" und commiten	(Letzter Schritt – der eigentliche Erstellvorgang)

## Schritt 1: Finden der benötigten Schriftarten

Die DocBook-Stylesheets unterscheiden zwischen sechs Schrift„familien“: für body, title, monospaced, symbol, dingbat und „sans“ (= sans-serif) text. Sie sind wie folgt definiert:

```
body.font.family      = serif
title.font.family    = sans-serif
monospace.font.family = monospace
symbol.font.family   = Symbol,ZapfDingbats
dingbat.font.family  = serif
sans.font.family     = sans-serif
```

Die `sans.font.family` wird in der Praxis nicht verwendet. Sie wird nur der Vollständigkeit halber aufgelistet. Die `symbol-` und `dingbat-`Familien werden vermutlich nicht verändert. Das bringt uns dazu, passende Fonts für `body`, `title` und `monospace` zu finden.

Für englische und andere westliche Sprachen wird „serif“ zu Times New Roman im PDF, „sans-serif“ zu Helvetica/Arial und „monospace“ zu Courier. Diese Schriftarten, ebenso der Symbol-Fonts, werden von allen Adobe PDF-Readern unterstützt. Deshalb müssen westliche Docwriter keine speziellen Anstrengungen unternehmen, damit ihre Sprache korrekt dargestellt wird. Die meisten nicht-westlich orientierten Zeichen sind hingegen nicht Teil der Adobe Standardfonts. Wenn Sie die Schriftarten nicht selbst einbetten, werden die Tools das PDF ohne Meckern erstellen, die unbekanntenen Zeichen werden jedoch durch ein „#“ ersetzt. Ihr Text sieht dann möglicherweise so aus: ##### ## ##### ##### ##### ## ## #####.

Sie sind gut beraten, Schriftarten zu verwenden, die allgemein auf Computersystem vorhanden sind, zumindest innerhalb Ihrer Lokalisierung. Metrics und Konfigurationsdaten werden ins CVS committet und sind durch andere Docwriter nutzbar, jedoch nicht die Schriftarten selbst. Diese müssen auf dem System des Benutzers vorhanden sein, wenn ein PDF erstellt werden soll. (Hinweis: Nur beim *Erstellen*, nicht beim *Lesen*!)

Wenn Ihre Sprache oder Schriften keinen Unterschied zwischen serifen- und nicht-serifen-Fonts macht, wählen Sie die gleiche Schriftart für den `body` und die `title-`Familien – oder verwenden Sie eine Auswahl, die Ihrer Sprache angemessen ist. Versuchen Sie jedoch nicht einen anderen Font für die `monospace-`Familie zu verwenden, auch wenn der Unterschied zwischen proportionalen und nicht-proportionalen Schriftarten in Ihrem Fall bedeutungslos scheint. Monospaced wird häufig verwendet, um Wörter in Sätzen hervorzuheben.

Jeder gewählte Font sollte diese Schrifteffekte unterstützen: normal, italic (kursiv), bold (fett), und bold italic (fett-kursiv). (Nur wenn dies für Ihre Sprache relevant ist.) Manchmal werden diese Unterschiede innerhalb einer Font-Datei vorgehalten, manchmal sind sie über bis zu vier Dateien verstreut. Nutzen Sie nur Type 1- und TrueType-Fonts.

Aufgrund der erheblichen Unterschiede zwischen den weltweit vorhandenen Sprachen und Schriften, ist es unmöglich eine spezifische Anweisung diesbezüglich zu geben. Fühlen Sie sich frei, jederzeit Probleme oder Fragen diesbezüglich in der `firebird-docs-`Liste zu diskutieren.

Sobald Sie sich für die zu verwendenden Fonts entschieden haben, notieren Sie sich die Ablageorte der Dateien. TrueType-Dateien besitzen die Erweiterung `.ttf`, TrueType-Sammlungen `.ttc`. Type 1-Dateien besitzen die Erweiterung `.pfb` (der Font selbst) und `.pfm` (die Metric-Info). Zum Einbetten von Type 1-Fonts, benötigen Sie beide Dateien, die `.pfb` und die `.pfm`.

## Schritt 2: Überschreiben der Stylesheet-Schriftarten

Jede Font-Konfiguration wird im Verzeichnis `manual/config/xx` platziert, wo `xx` Ihrem Sprachcode entspricht. Wenn das Verzeichnis bisher nicht existiert, erstellen Sie es. Bearbeiten Sie die Datei `fo-params.txt` in diesem Verzeichnis (wenn sie noch nicht existiert, kopieren Sie sie aus `manual/config`). Angenommen Sie arbeiten an einem japanischen Setup und Sie haben folgende Schriftarten gewählt: `MSGothic` für `titles`, `MSMincho` für normal `body text` und `WPJapanese` um `monospace` zu ersetzen. Sie würden dann die relevanten Teile der `manual/config/ja/fo-params.txt` wie folgt bearbeiten:

```
body.font.family=MSMincho
title.font.family=MSGothic
```

```
monospace.font.family=WPJapanese
```

Stellen Sie sicher, dass Sie jede veränderte Zeile unkommentieren, wenn notwendig.

Sobald die Zwischenausgabe der XSL-FO für japanisch erstellt wurde, wird das Dokument Referenzen für diese Schriftarten enthalten, statt der Standardfonts der Stylesheets. Ein Teil der `.fo`-Datei könnte folgendermaßen aussehen:

```
<fo:block keep-together="always" margin-left="0pc"
  font-family="MSMincho,Symbol,ZapfDingbats">
  <fo:block font-family="MSMincho" font-weight="bold"
    keep-with-next.within-column="always"
    space-before.minimum="0.8em" space-before.optimum="1.0em"
    space-before.maximum="1.2em"
    color="darkblue" text-align="start">
    <fo:block font-size="19.8pt">...Japanese text here...</fo:block>
  </fo:block>
</fo:block>
```

Die nächsten zwei Schritte behandeln den folgenden Aspekt: Die eigentliche Erstellung des PDFs.

## Schritt 3: Erstellung der Metric-Dateien

Für jeden verwendeten nicht-Standardfont, muss eine Datei mit Font-Metric-Informationen erstellt werden. Apache FOP benötigt diese Infos während der XSL-FO -> PDF-Konvertierung. Sie können die Metric-Dateien mit den `tlmetrics` und `ttfmetrics` targets Ihres Buildsystems generieren. Rufen Sie die Befehlszeile auf und wechseln Sie in das Verzeichnis `manual/src/build`. Geben Sie folgendes Kommando ein (alles in einer Zeile!):

```
build ttfmetrics
-Dff=D:\Path\To\fontfile.ttf -Dmf=filename.xml -Dsfxx
```

Um die TTF-Metric-Datei zu erstellen, oder

```
build tlmetrics
-Dff=D:\Path\To\fontfile.pfm -Dmf=filename.xml -Dsfxx
```

für eine Type 1-Metric-Datei.

Bitte beachten Sie:

- Sie müssen den kompletten Pfad zu Ihrem Font angeben, jedoch nur einen Dateinamen für die Metric-Datei. Letztere wird ersetzt in `manual/config/xx`, mit `xx`, Ihrem Sprachcode.
- Wählen Sie einen Dateinamen für Ihre Metric-Datei. Sinnvollerweise sollte der Name jedoch so gewählt werden, dass er Ihrem Font zugeordnet werden kann.
- Für das `tlmetrics` target, müssen Sie die `.pfm`-Datei verwenden, nicht die `.pfb`.
- Das zweite Zeichen von `tlmetrics` ist die Zahl 1 (eins), nicht der Buchstabe `e1`.
- Alternativ zum Parameter `sfxx` können Sie auch folgendes eingeben: `-Dmf=xx/filename.xml`.

Bitte berücksichtigen Sie, dass Sie diesen Schritt für jede Schriftart wiederholen müssen, die Sie hinzufügen. Wenn die fett- und/oder kursiv-Variationen in verschiedenen Dateien stecken, müssen Sie auch je eine eigene Metric-Datei erstellen.

### Wichtig

Metric-Dateien, die mit FOP 0.20.5 erstellt wurden (was Teil des Build-Tools bis April 2007 war), sind nicht mit der aktuellen FOP-Version (0.93 or higher) nutzbar. Wenn Sie solch alte Dateien in Ihrem Verzeichnisbaum besitzen, regenerieren Sie diese mit der jüngsten Tool-Version. Bitte beachten Sie, dass Durchführung eines CVS-Updates, die aktuellen Metric-Dateien beinhalten kann.

## Schriftarten-Sammlungen

Einige TrueType-Fonts werden in `.ttc`-Dateien gepackt (TrueType collections). Das `ttcmetrics` target ermöglicht Ihnen die Metric-Dateien für diese Fonts folgendermaßen zu erstellen:

```
build ttcmetrics -Dcf=D:\Path\To\collection.ttc
                 -Dfn=fontname -Dmf=filename.xml -Dsfxx=xx
```

Der `ff` (font file) Parameter wurde durch `cf` (collection file) ersetzt. Zusätzlich gibt es den `fn` (font name) Parameter. Um herauszufinden, welche Fonts in einer Sammlung existieren, rufen Sie das `ttcmetrics` target mit nur dem ersten Parameter auf, wie folgt:

```
build ttcmetrics -Dcf=D:\Path\To\collection.ttc
```

Die Ausgabe wird in einer Ausnahmebehandlung (exception) und einem 20 Zeilen langen Java Stack Trace resultieren. Vorher jedoch, finden Sie eine Liste aller fonts, die die Sammlung beinhaltet.

## Schritt 4: Erstellung einer FOP-Benutzerkonfigurationsdatei

Dies ist der komplizierteste Schritt. Sie müssen die `fop-userconfig.xml`-Datei bearbeiten, um FOP zu sagen:

- welche Schriftarten (und Variationen) eingebunden werden;
- wo die Schriftarten gefunden werden können;
- wo die Metric-Dateien gefunden werden können.

Dann fangen wir mal an:

1. Wenn notwendig, kopieren Sie die `fop-userconfig.xml` aus dem allgemeinen `manual/config`-Verzeichnis in Ihr Sprachverzeichnis (e.g. `manual/config/ja`).
2. Öffnen Sie die Datei in einem Text- oder XML-Editor und suchen Sie nach dem Eintrag `font-base`. Ersetzen „`xx`“ durch Ihren Sprachcode, so dass die URL auf das korrekte Konfigurationsunterverzeichnis zeigt. Entfernen Sie das Kommentarzeichen!
3. Wechseln Sie nun zum `<fonts>`-Element. Sie werden einige auskommentierte Beispiele finden.

4. Fügen Sie ein `<font>`-Element für den ersten Font hinzu:

```
<font metrics-url="msmincho.xml" kerning="yes"
      embed-url="file:///D:/WINNT/Fonts/MSMincho.ttf">
  <font-triplet name="MSMincho" style="normal" weight="normal"/>
</font>
```

Hinweise:

- `metrics-url` zeigt auf die Font-Metric-Datei, die Sie zuvor erstellt haben und die im gleichen Verzeichnis wie `fo-userconfig.xml` liegt.
- `embed-url` muss eine URL sein, die auf die Font-Datei selbst zeigt. Aber vorsicht! Für Type 1-Fonts, müssen Sie die `.pfb`-Datei angeben, nicht die `.pfm` wie Sie dies beim Erstellen der Metric-Datei taten.

### Wichtig

Selbst wenn das gesamte Setup bereits existiert, kann es notwendig sein, `embed-url` zu bearbeiten, da es möglich ist, dass die Schriftart bei Ihnen in einem anderen Verzeichnis liegt, als bei der Person, die den ursprünglichen CSV-Commit des Setups durchgeführt hat. Mehr Konfigurationsänderungen sollten nicht nötig sein.

- Das `font-triplet name` muss der gleiche sein, wie der Name, den Sie in `fo-params.txt` verwendet haben, um den Standardfont zu überschreiben.
5. Nun müssen Sie die Informationen für fett, kursiv und fett-kursiv hinzufügen. Häufig sind diese in verschiedenen Font-Dateien zu finden und Sie werden separate Metric-Dateien erstellt haben. Wenn dies tatsächlich der Fall ist, erhält jede Variation (fett, kursiv, fett-kursiv) ihren eigenen `<font>` Eintrag, zum Beispiel für fett-kursiv:

```
<font metrics-url="msmincho-bi.xml" kerning="yes"
      embed-url="file:///D:/WINNT/Fonts/MSMinchoBI.ttf">
  <font-triplet name="MSMincho" style="italic" weight="bold"/>
</font>
```

Beachten Sie, dass `font-triplet name` identisch für jede Variation sein muss: der Name, den Sie in `fo-params.txt` verwendet haben.

Manchmal existieren keine fett- oder kursiv-Varianten einer Schriftart. In diesem Fall müssen Sie dieses „faken“, da die Zwischen-`.fo`-Datei diese spezifizieren. Wenn die Zwischendateien nicht existieren, werden die gefürchteten `### ##### ##` in der PDF erscheinen. Für jede nichtexistierende Variation, muss ein `<font-triplet>`-Element hinzugefügt werden, für jede Variante, die stattdessen werden soll:

```
<font metrics-url="msmincho.xml" kerning="yes"
      embed-url="file:///D:/WINNT/Fonts/MSMincho.ttf">
  <font-triplet name="MSMincho" style="normal" weight="normal"/>
  <font-triplet name="MSMincho" style="italic" weight="normal"/>
</font>
```

Wenn weder fett noch kursiv existieren, erhalten Sie am Ende vier `<font-triplet>` Untereinträge im `<font>`-Element.

6. Wiederholen Sie die Schritte 4 und 5 für jede Font-Familie, die Sie zu `fo-params.txt` hinzugefügt haben.

Wenn alles erfolgreich war, sollten Sie nun in der Lage sein, PDFs in Ihrer Sprache zu erstellen..

### Wichtig

Die FOP 0.20.5-Benutzerkonfigurationsdateien (die bis April 2007 verwendet wurden) haben das falsche Format für unsere aktuelle FOP version (0.93 oder höher). Wenn FOP sich diesbezüglich meldet, führen Sie ein CVS-Update durch. Sollte das Dateiformat dann immer noch falsch sein, prüfen Sie die Datei `manual/config/fop-userconfig.xml` und geben Sie Ihrer sprachabhängigen Benutzerkonfigurationsdatei die gleiche Struktur.

## Schritt 5: Erstellung des PDF und Ausführung des Commits

Um Ihre Konfiguration zu testen, wird nun das PDF in Ihrer Sprache erstellt, z.B.:

```
build pdf -Drootid=qsg15-ru -Dsfx=ru
```

Untersuchen Sie das Ergebnis gründlich. Wenn Sie irgendwo im Dokument „### ##### ##“-Zeichen finden, wird Ihnen der Anzeigort Hinweise darauf geben, was schiefgelaufen ist:

- Erscheinen die Zeichen in den titles, haben Sie möglicherweise vergessen die `title.font.family` in `fo-params.txt` zu überschreiben und/oder die fett- oder fett-kursiv-Variationen in der Datei `fop-userconfig.xml` hinzuzufügen.
- Wenn Sie im body erscheinen, habe Sie vermutlich die `monospace.font.family` vergessen.
- Besteht das Dokument hauptsächlich aus „#“s, die titles sind hingegen OK, haben Sie wahrscheinlich die Angabe von `body.font.family` vergessen. (Dies sollte Ihnen die Schamesröte ins Gesicht treiben.)
- Wenn die Rauten an isolierten Stellen angezeigt werden, müssen Sie möglicherweise die Symbol- und/oder Dingbat-Familien überschreiben.

Vergleichen Sie Ihr PDF mit dem englischen Original, um hilfreiche Anhaltspunkte für die Probleme zu finden. Und natürlich gibt es da noch die `firebird-docs` list.

Sobald alles läuft, führen Sie einen Commit zum CVS aus:

- Das `manual/config/xx` Sprachverzeichnis (sofern dies noch nicht im CVS existiert).
- Die `fo-params.txt`-Datei in diesem Sprachverzeichnis.
- Alle `.xml`-Metric-Dateien im Sprachverzeichnis.
- Die `fop-userconfig.xml`-Datei im Sprachverzeichnis.

Wenn Sie keine Schreibrechte für das CVS besitzen, kontaktieren Sie ein Mitglied des Unterprojektes. Dieses wird Ihnen beim Commit helfen..

## Nachwort

Wir sind noch relativ "unbeleckt" bezüglich der Übersetzung der Firebird-Dokumente zu nicht-westlichen Schriften und ich glaube, wir haben noch viel aus der Erfahrung zu lernen. Bitte übermitteln Sie jegliche Fehler-

bericht, Kommentare und Vorschläge an die `firebird-docs`-List. Viel Glück bei Ihrer docwriting-Tätigkeit und Übersetzungen – Es ist großartig, Teil diese Unterprojekts zu sein!

## Anhang A: Dokumenthistorie

Die exakte Dateihistorie ist im manual-Module unseres CVS-Baumes aufgenommen; Siehe [http://sourceforge.net/cvs/?group\\_id=9028](http://sourceforge.net/cvs/?group_id=9028)

### Versionsgeschichte

0.1	22 Dez 2005	PV	Erste Edition.
0.1.1	23 Dez 2005	PV	Kleine Korrektur in Schritt 3-Abschnitt.
0.1.2	24 Dez 2005	PV	Inkorrekter Fontname-Anforderungen entfernt; Bemerkung über Bearbeitung der embed-file URL hinzugefügt..
0.1.3	25 Jan 2007	PV	<i>Wie wird das PDF erstellt:</i> Entfernung einiger Wörter aus dem letzten Satz, direkt oberhalb der Schritte-Tabelle.
1.0	18 Apr 2007	PV	<i>Erstelle Metric-Dateien</i> und <i>Erstelle eine FOP-Benutzerkonfigurationsdatei:</i> Auf aktuellen Stand mit FOP 0.93 gebracht. <i>Nachwort:</i> Wortführung geändert. Wir sind keine kompletten Newbies mehr :-)
1.0-de	21. Jul 2013	MK	<i>Deutsche Übersetzung basierend auf der englischen Dokumentenversion 1.0.</i>

## Anhang B: Lizenzhinweis

Der Inhalt dieser Dokumentation unterliegt der "Public Documentation License Version 1.0" (der „Lizenz“); die Dokumentation darf nur unter Respektierung dieser Lizenz genutzt werden. Kopien der Lizenz sind verfügbar unter <http://www.firebirdsql.org/pdfmanual/pdl.pdf> (PDF) und <http://www.firebirdsql.org/manual/pdl.html> (HTML).

Die Original-Dokumentation trägt den Titel *Using non-Western fonts in your Firebird docs*.

Der ursprünglich Autor der Original-Dokumentation ist: Paul Vinkenoog.

Copyright (C) 2005-2007. Alle Rechte vorbehalten. Kontakt zum Original-Autor: paulvink at users dot sourceforge dot net.

Übersetzung ins Deutsche: Martin Köditz.

Übersetzung ins Deutsche Copyright (C) 2013: Alle Rechte vorbehalten. Kontakt: martin dot koeditz at it-syn dot de.