Installation de DocBook sur un système Linux

Emmanuel Fleury

fleury@cs.auc.dk

Le but de ce document est d'expliquer comment installer aussi simplement que possible DocBook sur votre système (Mandrake, RedHat, Suse, Debian). Il donne une courte définition de DocBook et des termes qui gravitent autour (XML, SGML, ...) ainsi qu'un exemple rapide de document DocBook afin de tester votre installation.

1. Licence

Copyright © 2001 par Emmanuel Fleury

La copie, la distribution et/ou la modification de ce document est permise selon les termes de la licence de Documentation GNU version 1.1 ou ultérieure, publiée par la *Free Software Foundation* ; sans section invariante, ni texte de page de couverture, ni texte de quart de couverture. Une copie de cette licence peut-être trouvée ici (http ://www.fsf.org/copyleft/fdl.html).

2. Introduction

Au fur et à mesure des années, et au fur et à mesure que le nombre des formats de documents augmentait, il s'est fait sentir le besoin d'un langage de description de document qui permette d'écrire une seule fois un document et d'en tirer par la suite une version pour les formats les plus courants (ASCII, HTML, PostScript, PDF, ...). En effet, imaginez le travail que vous auriez si vous deviez écrire et maintenir une version HTML, LaTeX et ASCII d'un même document ! D'un autre coté, il existe actuellement des tas de manières de visualiser un document (papier, écran, PDA, braille, ...) et il serait un peu restrictif de n'en présenter qu'une.

DocBook est une tentative, plutôt réussie, dans ce sens. Il s'agit d'un langage de formatage de texte qui ne décrit que le contenu d'un document (sa sémantique) sans s'occuper de sa présentation. Une fois le document écrit on peut lui appliquer plusieurs '*filtres*' qui produisent divers formats (par exemple PostScript, PDF, HTML et ASCII).

Ce document décrit comment installer le langage DocBook sur un système Linux (RedHat, Mandrake, Suse, Debian). Ainsi que les principaux outils qui gravitent autour de ce langage (filtres permettant de produire des documents en divers formats, mode Emacs, *etc*).

Nous allons commencer par donner une rapide explication de ce qu'est réellement DocBook et de ses différentes versions (voir Section 3). Puis nous parlerons de l'installation proprement dites. D'abord sur des systèmes utilisant des paquetages RPM (voir Section 4), puis sur des distribution Debian (voir Section 5). Nous continuerons par un test de l'installation avec un petit exemple (voir Section 6). Et nous terminerons par quelques liens utiles (voir Section 7).

3. DocBook, sa vie, son oeuvre

DocBook est un langage à balises permettant de décrire le contenu de 'documents'. Par 'document', je veux dire un ensemble de textes, images, sons, vidéos mis ensembles avec une certaine cohérence. DocBook est donc un langage qui permet d'organiser et de classer ces différentes parties entre-elles suivant le sens que l'auteur veut leur donner.

Par exemple, je veux pouvoir organiser mon document en sections, sous-sections et sous-sous-sections. Il existe dans DocBook un ensemble de balises sect1, sect2 et sect3 qui permettent de faire cela.

Exemple 1. Organisation en sections d'un document

```
<sect1>
  <title>Mon titre de Section</title>
  <sect2>
    <title>Mon titre de Sous-section</title>
    <sect3>
      <title>Mon titre de Sous-sous-section</title>
      <para>Mon texte</para>
    </sect3>
  </sect2>
</sect1>
```

Les balises sect# ne font pas réellement partie du texte, mais permettent de donner la structure du document. Cette structure était autrefois donnée par la présentation que l'on donnait au document (les titres de section en gras et de taille plus importante que les titres de sous-sections, etc). Ici, on ne donne que l'information de la structure. On laisse à d'autres le soin de choisir la présentation à adopter quand à signifier le titre d'une section, sous-section ou sous-sous-section.

A de très rares exceptions près, vous ne verrez donc que des balises permettant de marquer la signification de chaque partie du document (sémantique) et non sa présentation. Cela permet évidemment de rendre possible la traduction du document ainsi décrit dans des formats aussi différents que l'ASCII, le PostScript, le HTML et le PDF.

À présent, puisque vous allez être amené à côtoyer le monde DocBook, vous allez souvent rencontrer d'étranges termes tels que XML, SGML, DTD, Schema, XSL, DSSSL, ... Je vais essayer d'expliquer rapidement à quoi correspondent ces termes.

En fait, chacune des balises DocBook sont elles-mêmes définies dans un autre langage plus général qui sert à définir des langages à balises. Ce langage est le SGML (*Standard General Markup Language*). À vrai dire, le SGML a aussi servi à définir HTML (c'est d'ailleurs en partie pour cela qu'il a été crée). Le problème, c'est que ce langage est très compliqué. Il comporte un nombre de possibilités qui sont souvent inusitées et qui obscurcissent sa compréhension. Le SGML a donc été simplifié et a donné le XML (*eXtendible Markup Language*). Ce dernier s'avère beaucoup plus simple à manier et permet de faire à peu près tout ce que l'on peut imaginer sur des définitions de documents.

Je sais que vous allez me dire : « C'est bien joli tout ça, mais comment ça marche ? »

Et bien, dans les deux cas (SGML et XML) le comportement est quasiment identique. Ils ont :

- Un langage pour définir des balises
  1. SGML : DTD (Document Type Definition)
  2. XML : DTD ou Schema
- Un langage pour définir la présentation
  1. SGML : DSSSL (Document Style Semantics and Specification Language)
  2. XML : XSL (eXtensible Stylesheet Language)

Ainsi, DocBook possède un DTD et un fichier de style (DSSSL ou XSL) suivant le format vers lequel vous voulez exporter votre document.

Là où les choses commencent à se compliquer un peu, c'est que DocBook, existe dans deux versions. L'une est écrite en SGML et l'autre est écrite en XML. Les différentes versions sont détaillées dans le Tableau 1.

Tableau 1. Les versions de DocBook

Langage	Versions DocBook
SGML	4.1, 4.0 et 3.1
XML	5.0alpha, 4.1.2, 4.1.1 et 4.1

Langage	Versions DocBook
XML Schema (OASIS)	4.1.2.3, 4.1.2.2, 4.1.2.1

En résumé, chaque distribution de DocBook est donc composée d'un fichier DTD et d'un ensemble de fichiers pour définir différentes apparences. Cependant, ces fichiers ne font que *définir* le langage et l'apparence. Il nous faut encore des logiciels qui soient capables de comprendre ces définitions et d'en tirer les formats voulus.

L'outil le plus couramment utilisé pour passer d'un format DocBook à un autre format est Jade (http ://www.jclark.com/jade/). Ce logiciel lit un fichier SGML, un DTD et un DSSSL et produit un document en résultat. La nouvelle mouture de Jade se nomme OpenJade (http ://sourceforge.net/projects/openjade/), mais reste encore expérimentale (notamment au niveau de la vitesse de traitement). Il existe aussi des logiciels permettant de produire des documents à partir de fichiers XSL, les plus communs étant SAXON (http ://users.iclway.co.uk/mhkay/saxon/), et COCOON (http ://xml.apache.org/cocoon/).

Toutefois, la version XML de DocBook étant encore expérimentale et les outils XML pas encore bien rodé, nous nous limiterons dans ce document à l'installation d'une version SGML et des outils qui vont avec.

4. Installer DocBook par RPM (RedHat, Mandrake, Suse)

Nous commencerons par le format de paquetage le plus courant, j'ai nommé : le RPM (RedHat Package Manager). La plupart des paquetages DocBook que proposent les distributions habituelles sont mal configurés et ne permettent pas de faire grand chose une fois installés. Je vous propose d'utiliser ceux conçus par le *KDE Documentation Project* (http ://i18n.kde.org/teams/en/). On peut trouver une documentation (http ://i18n.kde.org/doc/install/) (en anglais) qui détaille l'installation de ces paquetages. Je vous conseille de vous y référer si vous avez un problème car je me contenterai donner les grandes lignes de l'installation.

4.1. Téléchargement des paquetages

La première chose à faire est de télécharger les paquetages. Le site ftp officiel est ici (ftp ://ftp.kde.org/pub/kde/devel/docbook/RPMS/). Mais il est souvent surchargé et difficile d'accès. C'est pourquoi je vous conseillerai de consulter la liste des miroirs (http ://www.kde.org/mirrors.html) (le chemin est `/pub/kde/devel/docbook/RPMS/`).

Une fois que vous aurez choisi votre miroir, il vous faudra télécharger au moins les paquetages suivants :

- `sgml-common`: Installation de base du SGML (ressources et outils),

- `docbook-dtdxx-sgml`: DTD de DocBook. Le `xx` représente la version. Je vous conseille de prendre la version 4.1 qui est largement répandue. La 3.1 était la norme jusque récemment, mais celle-ci commence à s'essoufler,
- `docbook-style-dsssl`: Ce sont les fichiers DSSSL qui spécifient comment transformer les fichiers DocBook en formats lisibles (ASCII, PostScript, PDF, HTML, ...),
- `docbook-utils`: Ensemble de scripts permettant de manipuler plus simplement **jade**,
- `jade`: Le logiciel qui transforme vos fichiers DocBook en format lisible en suivant les fichiers DTD et DSSSL,
- `jadetex`: Un ensemble de macros nécessaires pour générer du TeX qui permet ensuite de générer du PostScript, du PDF et du RTF,
- `perl-SGMLSpm`: Support du SGML pour la langage Perl.

Je conseille aussi d'installer :

- `sgmltools-lite` : Un ensemble de macros pour manipuler jade plus simplement. Elles sont très utiles et incluent un petit convertisseur LinuxDoc vers DocBook,
- `psgml` : Un mode SGML pour Emacs. Le mode psgml est le mode SGML par défaut de XEmacs.

La plupart des distributions incluent ces paquetages dans leur CD-ROM d'installation. Il suffit de bien chercher. Toutefois, s'ils ne figuraient pas dans la liste des paquetages de votre distribution, je vous conseille de les chercher sur RPMfind (http ://rpmfind.net/linux/RPM/).

4.2. Faire le ménage

On va commencer par se débarrasser de tous les paquetages relatifs à SGML, DocBook et jade qui existent déjà afin d'éviter des conflits avec eux.

La première étape est de déterminer le nom de ces paquetages. Pour cela faites :

```
rpm -qa | grep jade
rpm -qa | grep docbook
rpm -qa | grep sgml
rpm -qa | grep perl-SGMLSpm
```

Il faut ensuite les enlever de votre système. Pour cela il faut faire pour chaque paquetage :

```
rpm -e <packagename>
```

4.3. Installation des paquetages

Une fois le ménage fait, nous pouvons passer à l'installation proprement dite. La commande pour installer un paquetage RPM est la suivante :

```
rpm -ivh <packagename>
```

Par exemple :

```
rpm -ivh sgml-common-0.3-1.noarch.rpm
```

Les paquetages doivent être installés dans l'ordre suivant :

1. `sgml-common`
2. `jade`
3. `jadetex`
4. `docbook-dtdxx-sgml`
5. `docbook-style-dsssl`
6. `perl-SGMLSpm`
7. `docbook-utils`
8. `sgmltools-lite`

Vous pouvez ensuite installer le mode Psgml pour Emacs si vous le désirez. Et voila, en théorie l'installation est finie à présent vous pouvez lire Section 6.

5. Installer DocBook sur une Debian

Les paquetages DocBook les plus fonctionnels pour Debian sont ceux qui sont sur la version dites '*sid*' qui est considérée à l'heure actuelle comme instable. Je considère ici que vous avez mis votre version de la Debian à jour vers la '*sid*'. Si ce n'est pas le cas, faites l'installation par RPM (Section 4).

Je suppose que vous utilisez **dpkg**. Je vais donc essayer de détailler l'ensemble des paquetages à télécharger, mais je vais sûrement oublier quelques dépendances. Alors ne me criez pas dessus et, au pire, corrigez de vous même cette erreur, et, au mieux, envoyez moi un petit correctif.

Évidemment, si, comme moi, vous utilisez **dselect** l'installation va être beaucoup plus facile. Mais je préfère ne considérer que **dpkg** pour éviter de léser les gens qui n'utilisent pas **dselect**.

5.1. Téléchargement des paquetages

Le meilleur endroit pour télécharger des paquetages Debian est bien évidemment le site Debian (http ://www.debian.org/). Là vous trouverez aisément la page consacrée aux paquetages (http ://www.debian.org/distrib/packages). Choisissez '*unstable*' et commencez la recherche sur les paquetages voulus. Voici la liste des paquetages dont vous avez besoin :

- `sgml-base`: Installation de base du SGML (catalogues et outils),
- `sgml-data`: DTD et entités,
- `sgmltools-lite`: Outils de conversion simplifiés pour utiliser **jade**,
- `docbook4.1`: Les DTDs de DocBook 4.1,
- `docbook-stylesheet`: Les fichiers DSSSL,
- `docbook-stylesheet-doc`: La documentation de DocBook (en anglais mais très utile),
- `jade`: Le logiciel qui transforme vos fichiers DocBook en format lisible en suivant les fichiers DTD et DSSSL,
- `jadetex`: Un ensemble de macros nécessaires pour générer du TeX qui permet ensuite de générer du PostScript, du PDF et du RTF,
- `psgml`: Mode SGML pour Emacs.

Une fois les dépendances résolues et les fichiers téléchargés, on peut passer à l'installation.

5.2. Installation des paquetages

dpkg étant un logiciel assez évolué, tout ce que vous avez à faire c'est de rassembler tout ces paquetages dans un répertoire et de lancer :

```
dpkg -i *.deb
```

Il se débrouillera tout seul pour résoudre les dépendances et les installer dans le bon ordre. Si par contre il vous manque un paquetage, il vous le signalera et vous devrez aller le chercher sur le site Debian. Si vous êtes suffisamment attentif, ce processus devrait pouvoir se terminer assez rapidement et vous pourrez allez voir Section 6.

6. Tester l'installation

Maintenant que tout est installé, nous allons voir comment ça marche ! L'Exemple 2 donne un fichier DocBook compilable. Il servira de fichier d'exercice pour cette section. J'appellerai ce fichier `example.sgml`. Dans chaque test, nous essayerons de générer des documents ASCII, HTML, PostScript puis PDF de ce document.

Exemple 2. Un exemple de document DocBook (`example.sgml`)

```
<!DOCTYPE article PUBLIC "-//OASIS//DTD DocBook V4.1//EN">

<article lang="fr">
<articleinfo>
  <title>This is a Test</title>

  <author>
    <firstname>Jean</firstname>
    <surname>Dupont</surname>
    <othername role="mi">L</othername>
    <affiliation>
      <address>
        <email>j.dupont@jdupont dot fr</email>
      </address>
    </affiliation>
  </author>

  <revhistory>
    <revision>
      <revnumber>v1.0</revnumber>
      <date>2001-05-30</date>
      <authorinitials>jld</authorinitials>
    </revision>
  </revhistory>

  <abstract>
    <para>Ceci est un test de document DocBook.</para>
  </abstract>

</articleinfo>

<sect1 id="test1">
<title>Test 1</title>
<para>
Test section 1.
```

```
</para>
  <sect2>
    <title>Test 1.1</title>
    <para>
        Test section 1.1
    </para>
  </sect2>

  <sect2>
    <title>Test 1.2</title>
    <para>
      <screen>
        -- Test section 1.2
        sgmltools -b ps test.sgml
      </screen>
    </para>
  </sect2>

</sect1>

<sect1 id="test2">
<title>Test 2</title>
<para>Test section 2.</para>

  <sect2>
    <title>Test 2.1</title>
    <para>
      Test section 2.1
    </para>
  </sect2>

  <sect2>
    <title>Test 2.2</title>
    <para>
      Test section 2.2
    </para>
  </sect2>

</sect1>
</article>
```

6.1. Les SGMLtools-lite

La commande **sgmltools** est extrêmement simple à utiliser. La syntaxe est toujours :

```
sgmltools -b <backend> example.sgml
```

En sachant que le '*backend*' correspond au format dans lequel on veut avoir le document. Il existe un certain nombre de '*backend*' dont voici la liste et une courte explication :

- **dvi**: Produit un document au format DVI.
- **html**: Produit un document au format HTML.
- **jadetex**: Produit un document à un format lisible par **jadetex**.
- **ld2db**: Transforme un format LinuxDoc en un format DocBook (très utile pour convertir vos anciennes documentations).
- **pdf**: Produit un document au format PDF.
- **ps**: Produit un document au format PostScript.
- **rtf**: Produit un document au format RTF.
- **txt**: Produit un document au format ASCII.

Exécutez donc les commandes suivantes pour produire (dans l'ordre) de l'ASCII, du HTML, du PostScript et du PDF :

```
sgmltools -b txt example.sgml
sgmltools -b html example.sgml
sgmltools -b ps example.sgml
sgmltools -b pdf example.sgml
```

Si tout s'est bien passé, l'installation est finie !

7. Références

Voici une petite liste non exhaustive de liens qui permettront d'approfondir vos connaissances en DocBook et de connaître un peu les à coté de ce format.

- Sites DocBook

  — Le Projet DocBook (http://www.docbook.org/)

  — Oasis DocBook (http://www.oasis-open.org/docbook/)

Un organisme indépendant qui participe à DocBook.

- Autres documentations

  — DocBook: The Definitive Guide (http://www.oreilly.com/catalog/docbook/chapter/book/docbook.html) [en]

  Livre O'Reilly sur DocBook (mis gracieusement en consultation en ligne). C'est l'un des documents les plus complet que l'on puisse trouver sur le Web. Il traite par défaut de la version 3.1 mais possède en annexe les différences des versions ultérieures.

  — DocBook Install mini-HOWTO (http://www.linuxdoc.org/HOWTO/mini/DocBook-Install/) [en]

  HOWTO détaillant comment installer DocBook et ses outils associés (Jade, SGML-lite, ...) à partir des sources (À déconseiller aux débutants).

  — LDP-Author Guide (http://www.linuxdoc.org/LDP/LDP-Author-Guide/) [en]

  Le guide du Linux Documentation Project à propos de la façon de rédiger des HOWTOs et des Guides en DocBook. Cette documentation contient à la fois des indications sur l'installation de DocBook ainsi que sur son utilisation.

  — Writing Documentation Using DocBook (http://www.caldera.de/~eric/crash-course/HTML/) (A crash Course) [en]

  Le guide du *KDE Documentation Project* à propos de la façon de rédiger des documentations en DocBook.

  — Article sur DocBook (http://lwn.net/2000/features/DocBook/) [en]

  Article sur DocBook de Linux Weekly News. Intéressant car il détaille les mécanismes de création d'un document.

- XML, SGML

  — Le W3C (http://www.w3.org/)

  Organisme qui gère les normes sur Internet (normes, recommandations).

  — OASIS (http://www.oasis-open.org/)

  Organisme qui joue un rôle important dans la définition de la norme XML.

  — XML-Pocket Reference (http://www.oreilly.com/catalog/xmlpr/) [en] [fr]

  Le XML-Pocket Reference est une *excellente* introduction à XML. Il contient pas mal d'exemples et permet de démarrer assez rapidement. Il peut aussi servir d'aide mémoire grâce à son format. Il manque néanmoins un index correct et des références sur les logiciels qui gravitent autour de XML. Une deuxième édition (http ://www.oreilly.com/catalog/xmlpr2/) est parue en anglais.

8. Remerciements

Je tient à remercier les gens de Linux-France et plus particulièrement :

- *Nat Makarévitch* pour avoir fondé Linux-France et m'avoir invité à participer à cette initiative,
- *Gilles Lamiral* pour sa participation active à Linux-France,
- *Etienne Herlent* pour avoir eu le premier le courage de vérifier la validité de ce document, corrigé les quelques erreurs des RPMs du KDE Documentation Project et éradiqué les typos de ce document,
- Et tous les autres participants de la liste de diffusions `mlfo`.