

The logo for Tirème SARL, featuring the company name in white, uppercase letters on a dark blue rectangular background.

DocBook, DocBook

Rédaction : Pierre Attar

Recommandation(s) liée(s) : [DITA](#)

[DocBook](#) est un modèle de documentation technique actuellement maintenu par un comité technique du consortium *OASIS Open*. Son utilisation s'avère particulièrement intéressante dès lors qu'il s'agit de créer des documents techniques ou des articles.

Cette recommandation est issue d'utilisateurs de l'industrie de l'informatique et de l'électronique. Du coup, il existe beaucoup de structures permettant de coder des informations liées à ce domaine : qu'il s'agisse des messages de spécification d'interface utilisateur ou d'objets et autres classes liées à une documentation issue d'une modélisation objet.

Les modèles issus de [DocBook](#) sont de conception extrêmement moderne et permettent d'ajouter ses propres éléments dans le modèle de base (la recommandation parle alors de "*wrapper*"). De même, si des objets ne sont pas utiles, ils peuvent être aisément supprimés. Beaucoup d'industriels se sont donc emparés de ce modèle et l'ont adapté à leurs propres besoins.

*Note. Certains trouvent que [DocBook](#) est trop difficile à paramétrer et, par conséquent, inutilisable. Les concepteurs travaillent alors aujourd'hui sur une vision simplifiée de [DocBook](#) appelée "*Simplified DocBook*".*

Toujours d'un point de vue conception, le modèle est *disponible* et *extrêmement bien documenté* sur Internet. Mais aussi, sont disponibles des *feuilles de styles*, utilisant [XSLT](#) et [XSL](#), destinées à faciliter la diffusion de ce type d'informations sur le *Web*, sur *Wap*, et sur *papier*.

Quelle sera la prochaine version de [DocBook](#) ? C'est un débat qui agite en ce moment la communauté *OASIS Open*, entre les "pro" de la spécification actuelle et ceux qui veulent, soit en faire quelque chose de plus modulaire quant à sa mise en œuvre, soit quelque chose de plus indépendant de l'industrie de l'informatique et de l'électronique, soit, enfin, quelque chose se rapprochant de [DITA](#) et de l'organisation de la documentation en modules autonomes d'information.

Objectifs

[DocBook](#) est le modèle de documentation technique actuellement maintenu par un comité technique du consortium *OASIS Open*.

[DocBook](#) a maintenant plus de dix ans. À l'origine créé par l'éditeur O'Reilly, son objectif était de simplifier l'échange de documentations UNIX. Puis, avec la création du groupe Davenport, son objectif s'est élargi, pour devenir le format d'écriture et d'échange de documentations techniques issues du monde des industries de l'informatique et de l'électronique.

La vision des documentations proposées par [DocBook](#) est proche du livre. Un document [DocBook](#) est soit un ensemble de livres, soit un livre seul, voire un seul article. Dans ce livre, tout peut être défini : qu'il s'agisse des Métadonnées de gestion comme les copyrights ou noms d'auteurs ; que ce soit aussi l'organisation des contenus par sections, avec des annexes ; qu'il s'agisse, enfin, des informations d'accès que représentent les tables des matières et autres index.

Dans les contenus eux-mêmes, le modèle différencie le niveau des composants (*component*) de celui des objets apparaissant directement dans le texte. Pour les deux types d'éléments, il existe toujours des éléments de présentation et des éléments plus sémantiques, liés à l'industrie visée.

Qui utilise [DocBook](#) ? Tout d'abord la communauté UNIX (plus particulièrement aujourd'hui la communauté LINUX), pour ses manuels. Ensuite, ce sont aussi les industriels et surtout les secteurs de l'Après-Vente. Enfin, avec l'avènement de "simplified [DocBook](#)", une vision simplifiée de [DocBook](#), beaucoup s'emparent du modèle pour l'adapter à leurs besoins, dès lors qu'ils ont besoin d'un modèle "documentaire" de base pour commencer à travailler.

Ce qui fait le succès de [DocBook](#) est, certes, le fait d'exister et d'être soutenu par le consortium *OASIS Open*. C'est aussi le fait que beaucoup d'outils implémentent ce modèle, tant pour créer les documents (*Frame SGML*, *Epic*, *XMLmind*, *XML Spy*, *XMetaL*, etc.) que pour utiliser les données issues de ce modèle. Dans le cadre de l'utilisation, on citera en tout premier lieu les feuilles de styles disponibles en "open source" proposées par **Norman Walsh**, l'éditeur du standard au sein d'*OASIS Open* ; on citera aussi les différentes implémentations disponibles dans les produits libres et/ou commerciaux (*Frame SGML*, *Epic*, *XPP*, etc.)

Principes

Les différents éléments du modèle

Les éléments de [DocBook](#) peuvent être classés selon la hiérarchie suivante :

- . ensembles ;
- . livres ;
- . divisions ;
- . articles ;
- . sections.

À la différence des standards de documentation technique basés sur la réutilisation de fragments pour créer une publication (voir par exemple [S1000D](#)), [DocBook](#) contient toute l'information nécessaire à une documentation technique. Ainsi, et par exemple, un livre comprendra des informations d'identification formelle, tous les composants d'index permettant la navigation, une préface, des chapitres, des annexes, des glossaires et une bibliographie.

Bien sûr, tous ces éléments sont optionnels et c'est à l'utilisateur de choisir ce qu'il veut mettre en place. L'utilisateur peut être restreint dans ces choix et il faut alors paramétrer le modèle.

Aux niveaux les plus bas on trouve les notions de :

Blocs

Ce sont des éléments de type paragraphe, avec un interligne avant et après dès lors qu'ils sont composés.

On trouve parmi eux différentes catégories de listes, d'avertissements, de synopsis, de tableaux, de figures, d'exemples, de média comme la vidéo, etc.

On trouve aussi des éléments plus sémantiques comme ce qui permet de décrire, en ingénierie logicielle, des options et des paramètres d'une commande. Pour décrire des programmes, on trouvera ce qui permet d'identifier des fonctions, leurs arguments et valeurs de retour, etc ; il y aura de la même façon les notion de message, voire de procédure de maintenance.

La provision pour ce type d'élément est extrêmement vaste, d'où la nécessité, souvent, d'adapter [DocBook](#) à ses propres besoins, afin de ne pas être obligé de manipuler tous les composants [DocBook](#).

Éléments en ligne

Ce sont les éléments que l'on retrouve directement mixés avec les phrases, les mots et les caractères.

À nouveau, on y trouve des éléments de présentation comme la mise en relief, les citations, les abréviations et autres acronymes. On trouve aussi tout ce qui permet de gérer les renvois (à des sections, des tableaux, des médias, etc.).

À nouveau aussi, on trouve un certain nombre d'informations liées à l'industrie visée comme des objets d'identification d'interface utilisateur, d'objets d'un langage de programmation, d'un système d'exploitation, etc.

Avec cette provision d'éléments, il est possible de réaliser ses propres documentations techniques, dès lors que l'on appartient à l'industrie visée. Qu'en est-il alors des utilisateurs d'une industrie différente ? Qu'en est-il aussi des utilisateurs de cette industrie qui ont des besoins beaucoup plus limités ?

C'est là qu'intervient la force de [DocBook](#). En effet, le modèle est conçu pour qu'il soit possible soit d'ajouter de nouvelles structures, soit d'en supprimer. Les deux opérations se font par redéfinition et la méthode fonctionne tant lorsque l'on utilise les [DTD](#) que lorsque l'on utilise les [Schema](#).

Modèle simplifié

Si [DocBook](#) contient autant d'objets structurels, c'est qu'ils sont nécessaires pour atteindre tous les objectifs documentaires de l'industrie visée. On a par ailleurs dit qu'il était possible de simplifier l'utilisation de [DocBook](#) par redéfinition du modèle.

Pour prouver la faisabilité de la chose et pour proposer aux implémenteurs une solution finalisée simple, les membres du comité [DocBook](#) proposent un modèle dit simplifié.

Dans ce modèle, beaucoup d'éléments ont été supprimés pour ne conserver que le strict minimum, toujours dans le cadre d'un utilisation en documentation technique.

Ce travail a été réalisé dans les "règles de l'art", par redéfinition du modèle complet.

Pour réaliser sa propre application, il est alors possible de choisir parmi les possibilités suivantes :

- utilisation de tout [DocBook](#) ;
- utilisation du modèle simplifié ;
- redéfinition de [DocBook](#) ;
- redéfinition du seul modèle simplifié.

Après avoir travaillé en redéfinition sur les deux modèles, la question est de savoir s'il existe une préférence sur celui sur lequel s'appuyer. Le choix est uniquement lié à la complexité du modèle souhaité et à ce que l'on souhaite utiliser dans la provision d'éléments **DocBook**. Il faut donc, dans un premier temps choisir le cadre dans lequel on s'insère, puis le redéfinir.

Informations connexes

On notera l'implémentation de **DocBook** réalisée dans le cadre de *Tirème* pour mettre en place la technologie SCHEMADOC : un atelier de documentation de modèles issus d'**XML** et représentés sous forme de **Schema**, voire de **DTD**.

Recommandations(s)

 *DocBook XML*

Recommandation, version 4.5, du 01-10-2006

Document sur <http://www.docbook.org/specs/docbook-4.5-spec.html>

 *DocBook SGML*

Recommandation, version 4.2, du 17-07-2002

Document sur <http://www.oasis-open.org/docbook/sgml/4.2/>

 *"Simplified" DocBook*

Recommandation, version 1.0, du 19-11-2002

Document sur <http://www.oasis-open.org/docbook/specs/cs-docbook-simple-1.0.html>