

From Markup to Markdown

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 30 septembre 2026

<https://www.bortzmeyer.org/from-markup-to-markdown.html>

Auteur(s) : Jim Hall

ISBN n°978-8-8688-2930-7

Éditeur : Apress / Springer

Publié en 2026

Aujourd'hui, bien des gens croient que la seule façon d'écrire un texte (sans le faire générer par un LLM) est d'utiliser un logiciel de traitement de textes graphique comme Writer. Mais il existe une autre méthode, plus ancienne mais toujours valable, les langages de marquage. C'est pour cela que j'ai acheté ce livre <<https://link.springer.com/book/10.1007/979-8-8688-2931-4>>, qui retrace l'histoire de ces langages de marquage et vise surtout leur utilisation pour l'écriture de textes technique et scientifiques.

Je le dis tout de suite, j'ai été déçu. Le livre <<https://link.springer.com/book/10.1007/979-8-8688-2931-4>> est cher (un peu moins en EPUB ou PDF mais quand même) pour un contenu qui semble surtout fait de divers articles de blog mis bout à bout. Certains de ces articles sont très intéressants, parlant de sujets peu ou pas abordés dans la plupart des sources, mais d'autres sont vraiment minces (TeX et LaTeX expédiés en 16 pages, dont de longs copier-coller de la sortie de ces programmes très bavards, ce qui est tout à fait inutile). De même, l'auteur a choisi de publier l'intégralité des programmes d'exemple. Ces programmes sont une très bonne idée, ils rendent plus concrets les cas discutés mais les mettre dans le livre, plutôt que sur un site Web à part, était une mauvaise idée.

Le livre commence avec Runoff, ancêtre de ces langages de marquage. Vous plongerez dans l'archéologie avec ce système, créé « juste après que nous ayons cessé de chasser les mammoths avec des outils de pierre ». (Vous pouvez d'ailleurs lire sa très intéressante histoire sur le blog de l'auteur <<https://technicallywewrite.com/2023/08/06/runoff>>.) Runoff date de l'époque où les impressions sortaient d'une grosse imprimante à laquelle les utilisateurs n'avaient pas accès directement. Pour vous dire mon âge, j'ai travaillé avec un tel système à l'INRETS dans les années 1980 et il y avait encore des opérateurs dont le principal travail était de sortir les "listings" de l'imprimante et de les déposer dans des casiers au noms des utilisateurs. Runoff était écrit en MAD et vous avez du code source en MAD p. 14 à 16. Dans les pages suivantes, l'auteur ré-implémente Runoff en C. L'exercice est intéressant (et fournira sans doute des bonnes idées de travaux pratiques à bien des enseignants) mais est sans doute trop détaillé, surtout si vous connaissez déjà C. L'auteur dit que cette partie est lisible par quelqu'un qui

n'est pas programmeur, que tout est expliqué, ce qui est vrai mais me laisse quand même sceptique. C n'est pas le langage le plus lisible...

Puis Runoff évolue, des descendants apparaissent, l'auteur détaille nroff et même ses macros -ms. (Là encore, point personnel, j'ai écrit des documentations en -ms. Des RFC ont été écrits avec nroff, cf. RFC 2223¹, section 20.) Autant cette partie est importante historiquement, autant lister autant de macros -ms était à mon avis excessif.

Puis le monde change et les traitements de texte modernes apparaissent. (Mais sans jamais avoir remplacé les langages de marquage.) Le premier cité est évidemment WordStar. J'apprends dans ce livre qu'en plus de ses menus, il avait aussi des marques ("*dot commands*"), très inspirées de celles de nroff, comme les premiers navires à vapeur, qui gardaient mâts et voiles au cas où. On pouvait donc formater ses documents Wordstar avec des .PA et .HE inclus dans le texte (p. 155 à 156)...L'auteur décrit ensuite le format sous lequel WordStar enregistre ses documents et écrit un programme en C pour les lire. (Les logiciels de traitement de texte modernes ont un format bien trop complexe pour que cet exercice soit encore réaliste.) Parmi les points rigolos de ce format, le remplissage pour que tout fichier, même vide, contienne toujours un multiple de 128 octets, ce qui facilitait la tâche du programme de relecture. (Dans cette heureuse époque où on lisait en C des fichiers venus de l'extérieur sans se demander s'ils ne décencheraient pas une faille de sécurité latente.) Mais bon, là encore, avoir mis tout le code source dans le livre est une mauvaise idée. (Même remarque pour les affichages de déboguage, avec l'intégralité des octets de remplissage, p. 177.)

Le livre détaille aussi l'histoire de l'impression puisque, à l'époque, il n'était guère envisageable de lire un long texte sur l'écran (et du temps de Runoff, on n'avait même pas toujours un écran). Le papier était donc crucial, et le livre explique le fonctionnement d'une imprimante matricielle puis crée une émulation en logiciel (toujours en C) d'une telle imprimante, permettant d'afficher sur l'écran à quoi aurait ressemblé un texte imprimé sur une telle machine. S'il y a dans ce livre de nombreuses pages passionnantes sur des points peu connus, là encore, l'auteur détaille trop.

Enfin on en arrive à HTML mais la partie sur le Web contient vraiment des erreurs énormes. Dire (p. 231) que le premier navigateur Web date de 1994 alors qu'il a cinq ans de plus, par exemple. Ou ne pas citer le CERN une seule fois et prétendre que le Web a été inventé à l'UIUC, ce qui est grave. Sinon, sur le Web, notez qu'on peut parfaitement écrire des documents complexes, y compris des livres, joliment formatés avec les techniques du Web <https://1024.socinfo.fr/2024/11/1024_24_2024_059.html> comme le fait C&F Éditions <<https://www.site.sarahgarcin.com/epe/fr/projects/c-f-editions>>. L'auteur décrit aussi Markdown, une alternative légère à XML et HTML (RFC 7764), et développe un logiciel de conversion de Markdown vers HTML.

Le livre se termine par une présentation du système de gestion de documents XML DITA dont je ne vois pas trop l'intérêt, il ne semble pas faire grand'chose de plus que XInclude ou des techniques similaires comme les entités XML. Puisqu'on parle de XML, j'en profite pour me faire de la publicité et dire que mon livre <<https://cyberstructure.fr/>> a été écrit en XML <<https://www.bortzmeyer.org/livre-tech.html>>.

Le livre concerne surtout le passé, vous n'y trouverez pas des systèmes récents comme Typst <<https://www.bortzmeyer.org/typst.html>>. Les langages de marquage ne sont pas une survivance du passé mais un domaine très actif et très utilisé (par exemple pour ce blog <<https://www.bortzmeyer.org/blog-implementation.html>>).

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc2223.txt>