

Markdown comme syntaxe universelle

atelier Stretching numérique

Damien Belvèze

10/03/2022



avantages du markdown

syntaxe très simple à apprendre permettant d'éditer dans une grande variété de formats (wiki, odt, doc, pdf, html, LaTeX, ce qu'on appelle la versatilité.

Depuis 1984, ceux qui écrivent de la poésie et ceux qui écrivent du code, à quelques exceptions près, n'écrivent plus sur le même support numérique. (le traitement de texte Wordstar vendu avec la microinformatique de l'époque permettait auparavant d'écrire les deux avec le même logiciel)

essor du traitement de texte qui ne permet plus d'écrire du code / maintien parallèle des éditeurs de texte pour les langages informatiques ⁽¹⁾

Nous avons donc perdu progressivement la capacité d'écrire dans un langage interprétable par la machine

Un traitement de texte est fait pour vous faciliter certaines mises en page en prenant en charge (et invisibilisant les opérations sous-jacentes), cela vous fait gagner du temps jusqu'à un certain point

le fait de recourir à ces traitements de texte opaques (parce que propriétaires et parce que ne permettant pas d'accéder à la liste des opérations) constitue un danger pour écrire la science (risque d'erreur et manque de transparence du logiciel)

LaTeX : découplage du texte et de sa mise en forme. Permet une finesse d'édition plus importante que les traitements de texte. convient aux chercheurs en informatique pour qui l'édition est une affaire de programmation.

Aaron Swartz et John Gruber ont inventé le markdown plus simple à utiliser que la LaTeX (courbe d'apprentissage beaucoup moins grande)⁽¹⁾

Le HTML comprend un grand nombre de balises. La mise en page du texte peut être également enrichie au moyen du CSS. La force du Markdown résulte dans l'utilisation de balises peu nombreuses, simples à mettre en place et à mémoriser. Il devient aisé de réaliser un document dont on maîtrise la mise en page sur le Web, sans pour autant devoir apprendre le HTML. Plus encore, en conjonction de Pandoc qui est un moteur de conversion, il devient possible de produire aussi bien du HTML, que des pdf, des docx, des epub, et même des présentations de type PowerPoint. On ne produit alors ses données qu'une seule fois et on les exporte mises en forme automatiquement pour plusieurs supports.

Au lieu d'apprendre à utiliser des interfaces qui changent régulièrement avec les changements de version et n'engendrent qu'un savoir restreint à l'éditeur de texte, apprenons plutôt le langage homme-machine pour compiler du texte avec des instructions de mise en forme. L'une des raisons pour lesquelles écrire en markdown sur Obsidian peut être aussi une bonne introduction au code (de même que pour Twine fournit une introduction au langage Haskell)

voir également la frustration d'universitaires qui considèrent que les traitements de texte restent trop attachés à l'impression comme finalité ⁽²⁾

petite démonstration

conversions de base

Les fichiers qui seront utilisés au cours de l'atelier sont téléchargeables sur github

instructions pour le formateur ou la formatrice :

- cloner le dossier sur le bureau de son ordi
- ouvrir le fichier kanagawa.md
- montrer la structure, à quoi correspond le markdown (éléments de syntaxe : headers, italique, titre, préambule, caractères gras italiques)

conversion en PDF

Instructions pour le formateur ou la formatrice :

- montrer la versalité du markdown, faire d'abord une conversion en PDF avec Pandoc

```
pandoc kanagawa.md -o kanagawa.pdf
```

Note1 : pour la conversion en PDF, on a besoin d'une version de LaTeX sur son ordinateur. Pandoc est installable sans besoin d'avoir les droits sur sa machine, en revanche pour installer une version de LaTeX (même légère TinyTeX, on doit avoir les droits sur sa machine)

Note 2 : si beaucoup de conversions sont réversibles (de MD à ODT, de ODT à MD, de MD à DOC, de DOC à MD), ce n'est pas le cas de toutes : on ne peut pas convertir un fichier PDF en MD à moins d'utiliser des logiciels propriétaires comme ceux d'Adobe. Par ailleurs toutes les conversions ne sont pas réversibles au même niveau. LaTeX et HTML étant potentiellement plus riches que Markdown, des éléments de mise en forme peuvent se perdre en route. Les champs dans Word ou Writer dont les styles peuvent être modifiés ne sont pas toujours aisément appariables à la syntaxe Markdown.

Pour convertir un texte en markdown qui comporte des références bibliographiques en PDF, il faut ajouter à la commande pandoc :

- le filtre à utiliser pour parser les citations en bibtex (Citeproc)
- le chemin vers le fichier qui contient les références bibliographiques (en bibtex)
- le chemin vers le fichier qui contient le style bibliographique (en csl)
- l'outil à utiliser pour faire la conversion LaTeX vers PDF

cela donne :

```
pandoc markdown.md --bibliography=biblio/bibliographie.bib --csl=biblio/nature.csl --citeproc
```

conversion en open document avec un modèle

template

un modèle se constitue en travaillant les styles sur le traitement de texte (définir qu'un titre aura telle forme, telle police, un titre 2, telle taille, telle police, une image sera alignée, ancrée au texte, etc.) Quand le modèle est fait, la conversion est simple :

```
pandoc -s kanagawa.md --reference-doc lettre.odt -o kanagawa.odt
```

conversion simple en html

Conversion simple avec HTML

```
pandoc -s kanagawa.md -o kanagawa.html
```

l'argument -s(tandalone) permet de compiler le visuel et le document en html

conversion en HTML avec feuille de style

On va appliquer la feuille de style CSS suivante (essai.css dans un dossier css) au document en markdown pour faire une page web avec la forme souhaitée :

<style>

/ le fait de mettre le css à l'interieur de balises <style> permettra*

d'utiliser la commande pandoc suivante :

pandoc -s fichierdepart.md -H fichier.css -o fichier_destination.html

l'argument -H encapsule tout le css dans le header du document créé, avant le body, ce qui fait que le Css qui s'y trouve va être utilisé pour présenter

*l'ensemble du document */*

/ dimensionnement de la page à la taille de l'écran
couleur de fond */*

html, body{

background-color: #11191f;
width:100%;
height: 100%;
margin:0 auto;
color: #ffffff;
}

/ couleur, taille et alignement du titre 1 = titre du document en markdown */*

h1{

text-align:center;
font-size: 2rem;
color: #edf0f3;
}

/ couleur, taille et alignement du titre 2 */*

h2 {

font-size: 1.75rem;
text-align:center;

```

    color: #599fa6;
}

/* espace de la citation */

blockquote.style {
    font: 14px/20px italic Times, serif;
    padding: 8px;
    background-color: #141e26;
    color: #a65980;
    border-top: 1px solid #e1cc89;
    border-bottom: 1px solid #e1cc89;
    margin: 5px;
    background-position: top left;
    background-repeat: no-repeat;
    text-indent: 23px;
}

/* liste à puces, puces carrées */

ul{
    list-style: square;
}

</style>

```

Le script de conversion s'écrit de la manière suivante :

```
pandoc -s kanagawa.md -H css/essai.css -o kanagawa.html
```

Envoyer avec Git la page vers le site github (ajouter, commettre, pousser)

créer son premier document en markdown avec l'éditeur en ligne Stylo

éditeurs en markdown

On peut écrire en markdown avec des éditeurs de toutes sortes : emacs, Atom, R (on utilise alors une version de markdown qui s'appelle Rmarkdown). Le bloc-note de Windows suffit même si pouvoir bénéficier de la coloration syntaxique d'un éditeur de texte un peu plus élaboré est une aide appréciable.

L'éditeur va interpréter comme du texte en markdown tous les fichiers qui ont l'extension .md

Si dans un premier temps on ne souhaite pas télécharger d'éditeur de texte supplémentaire, ni pandoc, ni LaTeX et qu'on veut pouvoir utiliser ces logiciels en ligne, un éditeur en ligne a été conçu pour les chercheurs en SHS. Il s'agit de

Stylo

Pour utiliser Stylo, il faut se créer un compte sur Huma-Num (infrastructure de recherche française dédiée aux humanités numériques). Ce compte vous permettra d'ailleurs d'avoir accès à d'autres services de cette infrastructure. On peut aussi se connecter avec son compte Twitter, ORCID ou HAL

Editer avec Stylo



créer un nouveau document Pour ouvrir la console d'édition, cliquer sur "open" (en grisé en haut à gauche), quand elle gêne, la refermer ("close")

Mettre en forme le texte suivant en se basant sur la structure du document :
<https://damienbelveze.github.io/stretchnum/kanagawa.html>

```
[titre] la vague de Hokusai  
[date] 2022-03-14  
[sous-titre] atelier Markdown comme syntaxe universelle
```

```
[Titre 2] La grande vague de Kanagawa
```

```
[Citation] "les vagues  
lentement déferlent vers le rivage  
une lune à peine visible"
```

```
Le peintre Hokusai a peint cette estampe en 1830 ou 1831 pendant la  
Période [Edo : mettre en italiques] Cette oeuvre est la plus connue du peintre. La couleur  
y domine est le bleu de Prusse
```

```
Mettre ici une image de votre choix en suivant la syntaxe :
```

![légende de l'image](chemin vers le fichier de l'image)

[Titre 2] signification de cette estampe

Il s'agit d'une [scène réaliste : mettre en gras] mais avec un [aspect dramatique : mettre en gras]
La signification de l'image varie d'ailleurs selon qu'on est occidental
(sens de lecture : gauche à droite) ou japonais (sens de lecture
vertical puis de droite à gauche)

Pour un occidental, les barques se dirigent vers la droite et tentent de
fuir la vague Pour un japonais, les barques dont les rameurs rament à
contre-courant se précipitent vers le creux de la vague.

[Titre 2] Préfecture de Kanagawa (principales villes)

[liste à puces]

- Atsugi
- Ayase
- Chigasaki
- Ebina
- Fujisawa

Quand le texte a été mis en forme, faire un export en HTML5 et un export en
PDF

Bibliographie

1. Dehut, J. En finir avec Word ! Pour une analyse des enjeux relatifs aux
traitements de texte et à leur utilisation. *L'Atelier des Savoirs* (2018).
2. Vitali-Rosati, M. Les chercheurs en SHS savent-ils écrire ? *The Conver-*
sation (2018).