



# Markdown

Nicolas.Borboen@epfl.ch, EPFL -STI - IGM - LENI (Laboratoire d'énergétique industrielle)

*Markdown is a small markup language that allows you to write text (without tags) and convert it into HTML for publication on a website. Easy to read, easy to write.*

Markdown est un langage de balisage léger qui permet d'écrire simplement du texte et de le convertir en code HTML pour sa publication sur un site web. Facile à lire, facile à écrire.

## Fiche descriptive

<b>Markdown</b>		
<b>Domaine</b>		
◆ Bureautique, langage de balisage simple		
<b>Licence</b>	<b>langue</b>	<b>version</b>
◆ BSD	◆ anglais	◆ 1.0.1 (2004)
<b>Autres alternatives libres</b>		
◆ MultiMarkdown		
◆ reStrucutredText		
◆ Textile		
◆ AsciiDoc		
<b>Sites Web</b>		
◆ Projet et téléchargement: <a href="http://daringfireball.net/projects/markdown/">daringfireball.net/projects/markdown/</a>		
<b>Plates-formes supportées</b>		
◆ Toutes		

## Note sur l'article

Afin de démontrer la lisibilité de la syntaxe Markdown, l'article est volontairement publié dans le code-source Markdown et sans mise en page particulière. Si la lecture sous cette forme vous dérange, vous pouvez lire la version convertie en HTML sur le site du [Flash Informatique]([flashinformatique.epfl.ch/spip.php?article2629](http://flashinformatique.epfl.ch/spip.php?article2629)). Néanmoins, c'est la simplicité de rédaction (comparé à du code HTML par exemple) et la lisibilité de la source qui sont remarquables avec Markdown...

### Introduction

Markdown est un simple langage de balisage humain fondé sur l'expertise des conversions de texte brut de toute l'histoire de l'informatique. Il a été écrit par John Gruber, qui édite le site <http://daringfireball.net>, aidé d'Aaron Swartz (voir

encart). C'est un petit (18KB) logiciel libre, diffusé sous licence BSD et sa version actuelle, v1.0.1, est datée de 2004.

Markdown désigne également le script de conversion de ce langage vers de l'[HTML][1], `_Markdown.pl_`, initialement écrit en Perl 5.6.0 avant d'avoir été transcrit dans beaucoup d'autres langages (et notamment en module CPAN `Text::Markdown`).

Le but de ce langage est la rédaction de documents pour le Web et l'intérêt de Markdown réside dans la lisibilité du texte non interprété, proche d'un E-mail au format texte, qui peut être utilisé tel quel ou converti en [HTML][1] valide. Le principe appliqué à Markdown est « facile à lire, facile à écrire ».

Influencée par d'autres langages tel que [Setext][1], [atx][2], [Textile][3], [reStructuredText][4], [Grutatxt][5] ou [EtText][6], la plus grande inspiration de la syntaxe Markdown reste néanmoins la rédaction d'E-mails au format texte.

[1]: <http://docutils.sourceforge.net/mirror/setext.html>  
 [2]: <http://www.aaronsw.com/2002/atx/>  
 [3]: <http://textism.com/tools/textile/>  
 [4]: <http://docutils.sourceforge.net/rst.html>  
 [5]: <http://www.triptico.com/software/grutatxt.html>  
 [6]: <http://ettext.taint.org/doc/>

## Exemples pratiques

Voici quelques exemples succincts de la syntaxe de base. La documentation officielle se trouve sur le site de Daring Fireball (<http://daringfireball.net/projects/markdown/syntax>) et des "cheat sheets" sont disponibles sur <http://code.ahren.org/markdown-cheatsheet>, [http://scadahacker.com/library/Documents/Cheat\\_Sheets/HTML%20-%20Markdown.pdf](http://scadahacker.com/library/Documents/Cheat_Sheets/HTML%20-%20Markdown.pdf) et sur [http://blog.lib.umn.edu/crosb002/leadership/Markdown\\_Cheat\\_Sheet.pdf](http://blog.lib.umn.edu/crosb002/leadership/Markdown_Cheat_Sheet.pdf). Les possibilités sont plus étoffées, ces extraits choisis sont représentatifs d'un texte simple.

### ### Code

Pour commencer, le formatage permettant la mise en page des exemples ci-dessous.

Pour mettre du code dans une phrase (balise HTML `<code>`), il faut utiliser l'accent grave (`` ``) autour du texte. Les paragraphes de codes sont identifiés à l'aide d'une indentation avant chaque ligne (soit 4 espaces ou une tabulation). Le texte contenu entre les balises `<code>` et `</code>` n'est pas interprété par le parseur et rendu tel quel dans le fichier HTML.

### ### Titres

Il existe plusieurs moyens d'écrire des titres `<h1>` et `<h2>`:

```
Header 1
=====
```

```
Header 2
-----
```

L'utilisation de ``#`` permet également de marquer le niveau du titre (et les ``#`` finaux sont optionnels):

```
# Header 1 #
## Header 2
### Header 3 ###
```

## Markdown

### ### Accentuation

L'accentuation sur un mot, à l'aide de l'italique\_, peut se faire avec des ``astérisques`` ou des ``underscores``. Pour du texte **gras**, on utilise des doubles ``astérisques`` ou doubles ``underscores``. *Italique* et **gras** peuvent alors être combinés de la manière suivante: ``astérisques et ``underscores``.

### ### Listes

1. Premier niveau d'une liste ordonnée
2. Une autre entrée de la liste
  - \* Une sous-liste non ordonnée
1. Troisième entrée de la liste ordonnée (la numérotation s'incrémente tant qu'une entrée commence par un chiffre)
  1. Une sous-liste ordonnée
4. Encore une entrée de la liste principale du texte devant être aligné avec l'entrée parente.
5. Concernant les listes non ordonnées
  - \* les astérisques ``
  - le symbole moins (tiret) ``
  - + et le symbole plus ``fonctionnent de la même manière

### ### Blockquotes

Markdown prévoit des blocs de citation. De manière similaire aux E-mails, il faut utiliser le symbole `>` avant chaque ligne. Le résultat est la transformation du paragraphe dans une balise HTML ``<blockquote>``.

### ### Liens

Markdown pourvoit deux manières de faire des liens, au fil du texte ou par référence. De la première manière, le lien sera stipulé directement là où il sera affiché:

```
Ceci est [un exemple](http://exemple.com/
"Titre optionnel") de lien.
À l'inverse, un lien par référence sera écrit comme
suit:
```

```
Ceci est [un exemple][id] de lien par
référence.
Ensuite, et ce n'importe où dans le document, il
est nécessaire de définir la référence du lien de
la manière suivante:
```

```
[id]: http://exemple.com/ "Titre optionnel"
La seconde méthode est alors pratique si un lien
apparaît plusieurs fois dans le document, ou pour
regrouper tous les liens d'une énumération.
```

### ### Images

Bien qu'il soit assez difficile d'imaginer une syntaxe intuitive pour placer des images dans un document texte, Markdown propose toutefois une notation similaire à celle des liens.

```
![Alt texte](/lien/vers/img.jpg "Titre
optionnel")
Ou, de manière similaire aux liens par références:
```

```
![Alt texte][img]
Puis la référence elle-même, ailleurs dans le
document:
```

```
[img]: url/vers/image "Titre optionnel"
Attention, aucun attribut concernant la taille de
l'image ou son positionnement n'est prévu dans
le langage Markdown. L'utilisation d'image reste
relativement sommaire.
```

### ### Tableaux

Markdown ne fournit pas une syntaxe ad hoc pour les tableaux. En cas de nécessité, il est possible d'utiliser directement du code HTML, mais de base aucune balise Markdown ne permet l'équivalent de ``<table>`` et consorts.

### ### HTML

En cas de besoin il suffit d'insérer du code [HTML] dans le document Markdown. Les seules restrictions sont que les éléments de type blocs (`<div>`, `<table>`, `<pre>`, `<p>`, etc.) doivent être séparés du contenu environnant par des lignes vierges, et que les balises de début et de fin de

bloc ne doivent pas être indentées avec espaces ou des tabulations. Inversement, les éléments de type span peuvent être placés n'importe où dans un paragraphe Markdown.

### Comparaison de syntaxe

Souvent comparé à [BBCode] (utilisé sur les forums), ou à la syntaxe de [Wikipedia], le [wikitexte], Markdown est objectivement plus lisible et plus simple à écrire ([source](http://www.codinghorror.com/blog/2008/05/is-html-a-human-markup-language.html)):

```
### BBCode
[size=150]Lightweight Markup Languages[/size]
According to [b]Wikipedia[/b]:
[quote]
A [url=http://is.gd/gns]lightweight markup
language[/url]
is a markup language with a simple syntax,
designed
to be easy for a human to enter with a simple
text
```

```
editor, and easy to read in its raw form.
```

```
[/quote]
```

```
Some examples are:
```

```
[list]
```

```
[*]Markdown
```

```
[*]Textile
```

```
[*]BBCode
```

```
[*]Wikipedia
```

```
[/list]
```

```
Markup should also extend to [i]code[/i]:
```

```
[code]
```

```
10 PRINT "I ROCK AT BASIC!"
```

```
20 GOTO 10
```

```
[/code]
```

### ### Wikipedia

```
==Lightweight Markup Languages==
```

```
According to '''Wikipedia''':
```

```
:A [[lightweight markup language]]
```

```
is a markup language with a simple syntax,
```

```
designed
```

```
to be easy for a human to enter with a simple
```

```
text
```

```
editor, and easy to read in its raw form.
```

```
Some examples are:
```

```
* Markdown
```

```
* Textile
```

```
* BBCode
```

```
* Wikipedia
```

```
Markup should also extend to 'code':
```

```
<source lang=qbasic>
```

```
10 PRINT "I ROCK AT BASIC!"
```

```
20 GOTO 10
```

```
</source>
```

### ### Markdown

```
Lightweight Markup Languages
```

```
=====
```

```
According to Wikipedia:
```

```
> A [lightweight markup language](http://
is.gd/gns)
```

```
is a markup language with a simple syntax,
```

```
designed
```

```
to be easy for a human to enter with a simple
```

```
text
```

```
editor, and easy to read in its raw form.
```

```
Some examples are:
```

```
* Markdown
```

```
* Textile
```

```
* BBCode
```

```
* Wikipedia
```

```
Markup should also extend to _code_:
```

```
10 PRINT "I ROCK AT BASIC!"
```

```
20 GOTO 10
```

# Markdown

## Les avantages

Lisibilité et simplicité sont les principaux avantages de Markdown. Des grands noms d'Internet ([StackOverflow][], [GitHub][], [Reddit][]) ou encore [Tumblr][] l'utilisent comme syntaxe d'édition. La simplicité et la popularité de l'interpréteur (parseur) fait qu'il est présent par défaut dans bon nombre d'outils, des paquets d'un système d'exploitation jusque dans les plug-ins d'un [CMS][], en passant par la coloration syntaxique des éditeurs.

## Les inconvénients

Bien que tous s'accordent sur l'esprit et l'élégance de Markdown, certains trouvent que plusieurs fonctionnalités lui manquent. La syntaxe Markdown ne couvrant pas tous ces usages, des "flavors" sont apparues. Les plus connues sans doute [GitHub][github\_md\_flavor], [MultiMarkdown][] ou encore [Markdown Extra][php\_md\_flavor]. Par conséquent, on pourrait lui reprocher la diversité de ses forks ou le manque de suivi du projet depuis 2004.

## Le bilan

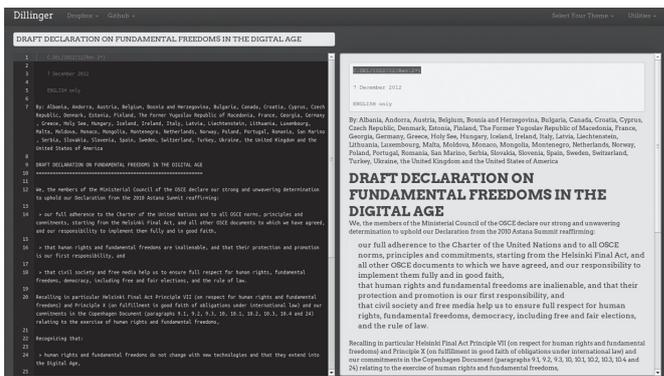
Le fait que des flavors existent prouve la popularité de la syntaxe Markdown et la reconnaissance qui lui est faite au travers des différents projets l'implémentant. Cependant, par choix je suppose, Markdown n'a jamais voulu se développer dans une direction qui aurait pu limiter les applications de base et qui rendrait le parseur plus complexe. Les flavors offrent de bons compléments, notamment la mise en forme de codes avec coloration syntaxique, tableaux, notes en bas de page, mises en forme de formules mathématiques, citations ou encore notes bibliographiques, mais à sa manière Markdown a su se contenter de sa fonctionnalité première: la mise en forme de documents simples (Read Me et autres documents .txt) et d'E-mails au format texte.

## Éditeurs

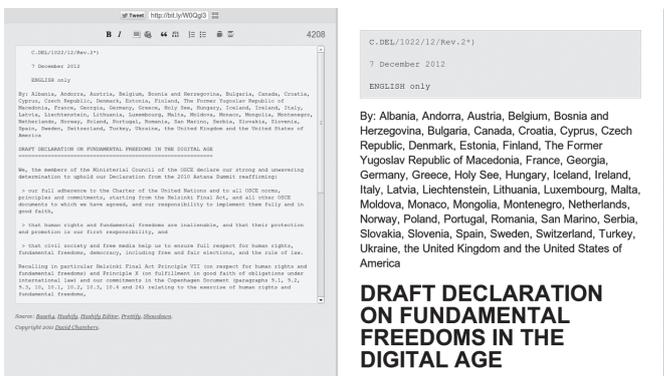
Bien qu'un simple éditeur de texte soit suffisant pour créer un document Markdown, certains éditeurs permettent de modifier la source et d'avoir un œil sur le résultat interprété en [HTML][] (live preview ou [WYSIWYG][]). Cela évite de devoir faire la conversion du fichier avec \_Markdown.pl\_ à chaque modification de la source et fournit un certain confort d'utilisation. On trouve ces logiciels pour toutes les plates-formes, mobiles compris, mais seule une sélection de ceux disponibles en ligne est citée ici, chacun étant libre de choisir ensuite une application compatible avec sa plate-forme. Tous ces éditeurs ont des particularités, les voici en quelques mots:

- \* [Dillinger.io](http://dillinger.io/) l'éditeur « web 2.0 » sous licence MIT avec numérotation des lignes, une multitude de thèmes, ainsi qu'importation et exportation sur [Dropbox][] ou [GitHub][].

[Dillinger.io markdown editor](dillingerio\_markdown\_edt\_50.png "http://dillinger.io/ markdown editor")

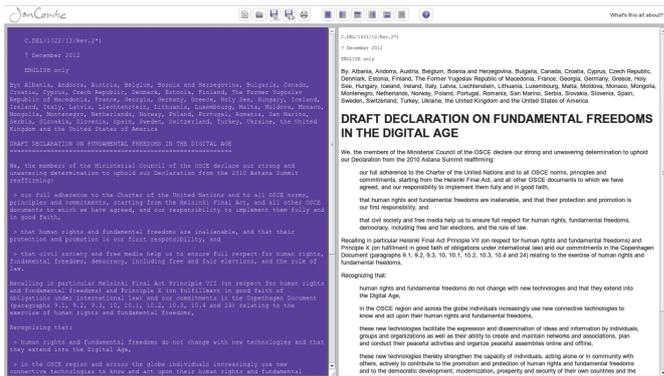


\* [Hashify](http://hashify.me/) est doté de quelques boutons facilitant l'édition, mais il fonctionne plutôt comme un [pastebin][] avec raccourcissement d'URL et génération de QR code. [Hashify markdown editor](hashify\_markdown\_edt\_50.png "http://hashify.me/ markdown editor")



\* [InstantMark](http://jraham.es/instantmark/) se différencie par un [CSS][] raffiné et la possibilité d'en changer (et donc de personnaliser à sa guise l'aperçu de son texte).

\* [JonCombe](http://joncom.be/experiments/markdown-editor/edit/) permet de choisir l'organisation de la fenêtre, partage horizontal ou vertical, etc. [Joncom.be markdown editor](joncome\_markdown\_edt\_50.png "http://joncom.be markdown editor")



\* [Pandoc](http://johnmacfarlane.net/pandoc/) est certainement le convertisseur le plus complet. Il permet de convertir du texte Markdown, reStructuredText, textile, HTML, DocBook, ou LaTeX vers une multitude de formats, dont ceux d'entrée plus quelques autres dont HTML5, AsciiDoc ou encore MediaWiki. Pandoc est écrit en [Haskell][] sous licence [GPL][] et est disponible en tant qu'exécutable pour toutes les plates-formes, permettant alors également la conversion vers PDF.

- \* [Pioul](http://markdown.pioul.fr/) est un éditeur minimaliste et astucieux, simple et efficace.
- \* [Showdown](http://softwaremaniacs.org/playground/showdown-highlight/) bien que d'apparence

## Markdown

simpliste, est le résultat du portage Javascript de `_Markdown.pl_` qu'un certain nombre de convertisseurs utilisent. Les sources de `_showdown.js_` sont disponibles sur <https://github.com/coreyti/showdown> (Licence BSD).

### MD2HTML et inversement

Le processus inverse, la génération d'un texte Markdown à partir d'un fichier HTML, est possible avec différents éditeurs dont `[html2txt]` (<http://www.aaronsw.com/2002/html2text/>) d'[\[Aaron Swartz\]](http://fr.wikipedia.org/wiki/Aaron_Swartz) ([http://fr.wikipedia.org/wiki/Aaron\\_Swartz](http://fr.wikipedia.org/wiki/Aaron_Swartz)), `[Markdownify]` (<http://milianw.de/projects/markdownify/>) (`[PHP]`) et en ligne avec <http://domchristie.github.com/to-markdown/> ou comme vu précédemment `[Pandoc]` (<http://johnmacfarlane.net/pandoc/try>).

### Conclusion

Un texte balisé avec Markdown pourra aussi bien être utilisé tel quel que transformé en HTML. Markdown simplifie grandement l'édition en évitant un balisage conséquent et permet à l'utilisateur de se focaliser sur le contenu du fichier. Les possibilités de Markdown sont minimalistes, son usage est limité à des documents tels que notes, Read Me ou E-mails au format texte. Les documents plus complexes ne conviennent pas à Markdown car ils seraient difficilement lisibles sans être transformés en HTML et plus compliqués à écrire. Par conséquent Markdown tient sa promesse « facile à lire, facile à écrire ».

### Pour plus d'informations

\* La page Markdown de Wikipedia: <http://fr.wikipedia.org/wiki/Markdown>  
\* L'article sur AsciiDoc écrit par Pascal Fabbri (FI3/2012) pour ceux qui auraient besoin d'un langage de balisage couvrant plus de fonctionnalités: <http://flashinformatique.epfl.ch/spip.php?article2527>  
\* Un article du blog de Florent Gallaire comparant des langages de balisage légers: <http://fgallaire.flect.net/comparaison-langage-balisage-markup-lightweight-leger-txt2tags-pandoc-docutils-asciidoc-deplate-stx2any-aft-markdown-textile/>  
\* Une liste non exhaustive d'implémentations: [http://en.wikipedia.org/wiki/List\\_of\\_Markdown\\_implementations](http://en.wikipedia.org/wiki/List_of_Markdown_implementations)  
\* Une comparaison de langages de génération de documents: [http://en.wikipedia.org/wiki/Comparison\\_of\\_documentation\\_generators](http://en.wikipedia.org/wiki/Comparison_of_documentation_generators)  
\* Une liste d'autres langages similaire à Markdown: [http://en.wikipedia.org/wiki/List\\_of\\_lightweight\\_markup\\_languages](http://en.wikipedia.org/wiki/List_of_lightweight_markup_languages).



Article du FI-EPFL 2013 sous licence CC BY-SA 3.0 / N. Borboën

## Aaron Swartz (1986-2013), activiste Internet et jeune prodige de l'informatique

Aaron Swartz, [fr.wikipedia.org/wiki/Aaron\\_Swartz](http://fr.wikipedia.org/wiki/Aaron_Swartz) est le fondateur de [demandprogress.org](http://demandprogress.org/), <http://demandprogress.org/> qui a lancé la campagne contre SOPA/PIPA, [fr.wikipedia.org/wiki/Manifestations\\_contre\\_SOPA\\_et\\_PIPA](http://fr.wikipedia.org/wiki/Manifestations_contre_SOPA_et_PIPA) aux États-Unis, soutenue par plus d'un million de membres. Auteur de nombreux articles, il est devenu membre du **Harvard Ethics Center Lab on Institutional Corruption**, [www.ethics.harvard.edu](http://www.ethics.harvard.edu).

Il a également développé le site **theinfo.org** et son article **Who Writes Wikipedia** a été largement cité. Actif dans les normes du Web et travaillant avec Tim Berners-Lee au MIT (2010-2011), Aaron a participé à l'élaboration de la spécification du format **RSS**, [fr.wikipedia.org/wiki/RSS](http://fr.wikipedia.org/wiki/RSS) dès l'âge de 14 ans et est devenu membre du **W3C**, [www.w3.org](http://www.w3.org) et du **RDF**, [www.w3.org/RDF/](http://www.w3.org/RDF/) Core Working Group.

En 2004, le créateur de Markdown John Gruber le remercie énormément pour ses commentaires, ses idées et ses tests sur la syntaxe Markdown. Aaron Swartz est le créateur du très pratique **html2text**, [www.aaronsw.com/2002/html2text/](http://www.aaronsw.com/2002/html2text/) permettant de transformer du code HTML en fichier Markdown.

Il est cofondateur de **Reddit**, [reddit.com](http://reddit.com) en 2005, qu'il développe avec son framework Python **web py**, [webpy.org](http://webpy.org) (également utilisé sur **Local.ch**).

Sensible aux questions d'accès au savoir et supporter de l'**Open Access**, [fr.wikipedia.org/wiki/Libre\\_accès\\_\(édition\\_scientifique\)](http://fr.wikipedia.org/wiki/Libre_accès_(édition_scientifique)), il a dirigé le développement de l'**Open library**, [openlibrary.org](http://openlibrary.org) en 2006. Il est aussi l'auteur de **Jottit**, <https://www.jottit.com> en 2007, un générateur de site Web ultra simplifié.

Le 19 juillet 2011, il est accusé d'avoir téléchargé près de 5 millions d'articles issus du système d'archivage de publications universitaires et scientifiques **JSTOR**, [www.jstor.org/](http://www.jstor.org/). Après la révélation de ses agissements (via le **blog Alex Stamos**, [unhandled.com/2013/01/12/the-truth-about-aaron-swartzs-crime/](http://unhandled.com/2013/01/12/the-truth-about-aaron-swartzs-crime/)), Aaron retourne ses disques durs (en promettant de ne pas les diffuser) et JSTOR décide alors de ne pas entamer de poursuites judiciaires. Néanmoins, le bureau du procureur et le MIT maintiennent leurs poursuites.

Le 11 janvier 2013, Swartz se suicide à l'âge de 26 ans par pendaison dans son appartement. Son procès fédéral en lien avec ces accusations de fraude électronique devait débuter en février. Il encourait une peine d'emprisonnement pouvant atteindre 35 ans et une amende s'élevant jusqu'à 1 million de dollars. Dépressif depuis plusieurs années, sa famille dénonce la pression du système judiciaire américain à son encontre.

Aaron Swartz s'efforçait de construire un monde meilleur et plus ouvert, d'ailleurs certains le définissent déjà comme un martyr du combat pour la libération de l'accès aux œuvres et au savoir. Plusieurs initiatives ont vu le jour à la suite de son décès: des chercheurs publient leurs travaux en accès libre en forme d'hommage à son engagement, sur le site Internet [PDFtribute.net](http://PDFtribute.net) et à travers le hashtag #PDFtribute sur Twitter.

Au génie d'Aaron, en espérant qu'il ne soit pas mort en vain !

■ [www.aaronsw.com](http://www.aaronsw.com)

■ [www.rememberaaronsw.com](http://www.rememberaaronsw.com) ■