



Création et conversion de documents avec Pandoc



Jean-Daniel.Bonjour@epfl.ch, responsable informatique et chargé de cours, ENAC

In a previous article [1] you've discovered Markdown, a lightweight document markup language. We present here Pandoc, a remarkable tool that greatly expands the possibilities of Markdown.

Un précédent article [1] vous a fait découvrir Markdown, langage de balisage léger de documents. Nous vous présentons ici Pandoc, un remarquable outil démultipliant les possibilités de Markdown.

Fiche descriptive

PANDOC		
Domaine d'utilisation		
<ul style="list-style-type: none"> ◆ Élaboration de documents et pages Web par balisage léger ◆ Conversion de documents dans de multiples formats 		
Licence	langue	version
◆ GPL v2	◆ anglais	◆ 1.11.1
Quelques alternatives libres		
<ul style="list-style-type: none"> ◆ Txt2tags (txt2tags.org) ◆ Docutils & reStructuredText (docutils.sf.net) ◆ AsciiDoc (www.methods.co.nz/asciidoc/) ◆ Deplate (deplate.sourceforge.net) 		
Sites Web		
<ul style="list-style-type: none"> ◆ Site de l'auteur: johnmacfarlane.net/pandoc/ ◆ Site du projet: code.google.com/p/pandoc/ 		
Plates-formes supportées		
◆   		

Introduction

On distingue fondamentalement deux catégories d'outils d'élaboration de documents. Les logiciels de **traitements de texte** WYSIWYG d'une part, très axés sur la mise en forme, intégrés à des suites bureautiques très lourdes dont on n'utilise souvent qu'une fraction des possibilités. Divers **langages de balisage** d'autre part (TeX/LaTeX pour les documents scientifiques, MediaWiki pour les wikis, HTML pour les pages Web, etc.), très riches, mais peu adaptés à un usage tel quel dans un éditeur de texte (difficulté de distinguer facilement contenu et balises ainsi que manipuler celles-ci...). C'est là que les langages dits de **balisage léger** [2, 3, 4] viennent à la rescousse : tout en offrant une richesse fonctionnelle suffisante, ils s'appuient sur une syntaxe naturelle facilitant

l'écriture sans entraver la lecture, permettant ainsi au rédacteur ou lecteur à se concentrer sur le fond.

Présentée dans un précédent article de ce journal par Nicolas Borboën [1], Markdown est une syntaxe de balisage très légère définie en 2004 par John Gruber [5]. Essentiellement orientée vers la rédaction pour le Web, elle est utilisée par d'importants sites Internet (GitHub, StackOverflow, Tumblr, Reddit...). Mais son auteur n'ayant depuis lors jamais souhaité enrichir cette syntaxe, des lacunes assez importantes limitent son usage lorsqu'il s'agit d'élaborer des documents plus riches (tableaux, listes évoluées, formules, notes de bas de page, table des matières...). Plusieurs alternatives au convertisseur `Markdown.pl` original de 2004 ont cependant vu le jour au cours de ces dernières années, étendant chacune à leur façon le standard Markdown. Parmi ces outils, **Pandoc** sort clairement du lot, s'affirmant aujourd'hui comme le plus complet et polyvalent, qu'il s'agisse du nombre de formats supportés ou de l'étendue de ses extensions à la syntaxe Markdown.

Qu'est-ce que Pandoc?

Laissons John MacFarlane, auteur principal de Pandoc (et Prof. de philosophie (!) à l'Université de Berkeley) définir lui-même ce qu'est Pandoc [6]:

«Pandoc est constitué d'une bibliothèque de conversion entre formats de balisage écrite en langage Haskell, ainsi que de l'outil en mode commande associé. Pandoc peut actuellement lire les formats Markdown (standard ou étendu), JSON, reST (reStructuredText), HTML, Textile, DocBook XML, Mediawiki et LaTeX ; il peut écrire les formats Markdown (standard ou étendu), plain-text, reST, Textile, XHTML, HTML5, LaTeX, ConTeXt, DocBook XML, OpenDocument XML, GNU Texinfo, Emacs Org-mode, AsciiDoc, RTF, ODT, docx, MediaWiki, FictionBook2, groff man pages, ePub (v2 et v3), PDF, ainsi que des slide shows HTML (S5, Slidy, Slideous, DZSlides) ou PDF (LaTeX beamer).

Il implémente un grand nombre d'extensions au standard Markdown, tout en offrant un mode de compatibilité qui lui permet d'être utilisé en lieu et place du convertisseur original `Markdown.pl`. Contrairement aux autres outils de conversion Markdown qui se basent sur des substitutions de type regexp, l'architecture modulaire de Pandoc consiste en un ensemble de modules de lecture (readers) qui interprètent le texte fourni dans différents formats et produisent une représentation interne native du document, et un ensemble de modules d'écriture (writers) qui exportent cette représentation dans différents formats cibles. Implémenter un nouveau format (d'entrée ou de sortie) équivaut donc à implémenter le module correspondant.»


```
~~~~~ { .python .numberLines startFrom="10" }
#!/usr/bin/env python3
from time import localtime
heure = localtime().tm_hour
if heure < 17:
    print("Bonjour !")
else:
    print("Bonsoir !")
~~~~~
```

```
10  #!/usr/bin/env python3
11  from time import localtime
12  heure = localtime().tm_hour
13  if heure < 17:
14      print("Bonjour !")
15  else:
16      print("Bonsoir !")
```

Listes numérotées

Les extensions implémentées par Pandoc dans le domaine des listes sont nombreuses et particulièrement bienvenues.

S'agissant des listes numérotées, elles ne sont pas limitées aux chiffres arabes, mais peuvent être de type alphabétique ou chiffres romains, minuscules ou majuscules. De plus le numéro spécifié pour le premier élément de la liste est significatif, indiquant à Pandoc que la numérotation doit débiter à cette valeur et non pas à 1. On peut en outre faire usage de parenthèses **()** en lieu et place du point pour indiquer qu'il s'agit d'une liste numérotée, et le caractère dièse **#** peut être utilisé comme substitut au numéro à partir du second élément de la liste.

```
11. Liste dont la numérotation démarre au numéro
spécifié
#. Élément suivant...

(A) Liste numérotée par des lettres majuscules
  f. sous-liste
  #. numérotée par des lettres minuscules
(#) Élément suivant...

i) Liste numérotée par des chiffres romains
minuscules
#) Élément suivant...
```

11. Liste dont la numérotation démarre au numéro spécifié
12. Élément suivant...

A. Liste numérotée par des lettres majuscules

f. sous-liste

g. numérotée par des lettres minuscules

B. Élément suivant...

i. Liste numérotée par des chiffres romains minuscules

ii. Élément suivant...

On peut créer une *liste référençable* et numérotée de façon continue à travers tout le document à l'aide du caractère **@**. Celui-ci peut être suivi d'un *label* permettant de se référer à l'élément considéré de la liste ailleurs dans le document. Ce type de liste peut être vu comme une alternative aux notes de bas de page présentées plus loin.

Dans l'exemple ci-dessous, notez que les **()** sont nécessaires, mais que les **[]** sont purement décoratifs (ne font pas partie de la syntaxe).

```
À l'EPFL [epfl], la faculté ENAC [enac]
forme des ingénieurs et architectes...
```

```
(epfl) École polytechnique fédérale de Lausanne
(enac) Environnement naturel, Architectural
et Construit
```

A l'EPFL [1], la faculté ENAC [2] forme des ingénieurs et architectes...

1. École polytechnique fédérale de Lausanne
2. Environnement naturel, Architectural et Construit

Listes de définitions

Les listes de définitions correspondent à la structure HTML suivante, un peu désuète, bien que fort pratique:

```
<d1>
<dt>terme</dt>
<dd>définition 1</dd>
<dd>définition 2</dd>
</d1>
```

Dans la syntaxe Pandoc, le `terme` doit tenir sur une seule ligne et peut être optionnellement suivi d'une ligne vide. Il doit être suivi d'une ou plusieurs *définition(s)*, chacune débutant par le caractère deux-points **:** ou tilde **~**, ce caractère sera facultativement précédé d'une ou deux *espaces*, mais obligatoirement suivi d'une à quatre *espaces* ou d'un *tab*.

```
**ENAC**
~ Environnement Naturel, Architectural et
Construit
~ Faculté comportant 4 instituts et offrant 3
plans d'études
```

ENAC

Environnement Naturel, Architectural et Construit
Faculté comportant 4 instituts et offrant 3 plans d'études

Notes de bas de page

Les notes de bas de page Pandoc sont numérotées de façon continue à travers tout le document. S'agissant d'une conversion en HTML, l'ensemble des notes seront rassemblées par défaut tout au bas de la page Web. Ces notes peuvent être définies de deux manières : par référence (permettant dans ce cas de s'y référer à plusieurs endroits dans le texte) ou inline (au fil du texte).

```
Définition de notes par référence[^noteRef] ou
de manière inline^[Texte de la seconde note]...
```

```
[^noteRef]: Texte de la première note
```

Suite...

Définition de notes par référence¹ ou de manière inline²...
Suite...

1. Texte de la première note↔
2. Texte de la seconde note↔

Tableaux

Pandoc offre pas moins de quatre techniques pour définir des tableaux. Nous illustrons ci-dessous celle du *tableau multiligne*. Nous vous renvoyons à la documentation [7] ou [8] pour davantage d'explications.

```
-----  
Ici aligné à gauche          Centré dans la colonne  
-----  
Contenu de cellule multi-ligne !  Usage de styles Markdown :  
                                     **gras**, _italique_, `code`  
-----  
: Légende du tableau...
```

Ici aligné à gauche	Centré dans la colonne
Contenu de cellule multi-ligne !	Usage de styles Markdown : gras , <i>italique</i> , <code>code</code>

Légende du tableau...

Bloc de titre et de métadonnées du document

Au tout début du document Markdown/Pandoc, donc sans insérer de contenu ou de lignes vides avant, il est possible de spécifier, au moyen de trois lignes débutant par le caractère pour-cent %:

1ère ligne: le titre du document, qui générera en HTML à la fois le `<title>` de la page et le premier titre `<h1>`,

2ème ligne: le nom des auteurs, qui s'afficheront en HTML dans un titre `<h2>` et seront insérés dans les balises `<meta name="author" ...>`,

3ème ligne: la date de révision du document, qui s'affichera en HTML dans un titre `<h3>`.

Les premières lignes de la présente page Web résultent du code Markdown ci-dessous:

```
% Création et conversion de documents avec Pandoc  
% <Jean-Daniel.Bonjour@epfl.ch> (EPFL-ENAC-IT) \  
% responsable IT et chargé de cours ENAC  
% Juillet 2013. [Flash informatique EPFL](http://  
% flashinformatique.epfl.ch/).
```

Autres possibilités offertes par Pandoc

Sans entrer davantage dans les détails, voici ce dont Pandoc est encore capable:

- lorsque l'on définit des **blocs HTML** (`<div>`, `<table>`...), la syntaxe Markdown est interprétée par Pandoc à l'intérieur de ceux-ci, contrairement à ce que fait le convertisseur `Markdown.pl` original;
- il est possible d'insérer du code **LaTeX**, **TeX** et **ConTeXt** dans un document Markdown, ce contenu étant passé tel quel par Pandoc lors d'une conversion en LaTeX et ConTeXt ;
- Pandoc permet la gestion de **citations** et références **bibliographiques** dans de nombreux formats (BibTeX/bib, MODS, RIS, ISI/wos, MedLine, Copac, JSON, EndNote enl/xml...) et s'interface avec l'excellent logiciel libre Zotero (www.zotero.org) ;
- Pandoc permet de produire très aisément des **slide shows HTML** (de type S5, Slidy, Slideous ou DZSlides) ou PDF (LaTeX *beamer*) ;
- Pandoc utilise un mécanisme de **templates** que l'on peut adapter pour changer la structure des fichiers de sortie ;
- l'auteur de Pandoc a en outre développé un Wiki nommé **gitit [10]** s'appuyant sur le backend Git et utilisant la syntaxe Markdown/Pandoc.

Si vous êtes intéressé par l'évolution de Pandoc, voire à participer à sa communauté de développeurs, rejoignez son forum de discussion [11].

Installation de Pandoc

Le projet Pandoc est très actif. Si l'on souhaite bénéficier des dernières fonctionnalités et extensions à Markdown, il est donc important d'utiliser la version la plus récente !

Sous Windows

Téléchargez l'installateur `pandoc-<version>.msi` depuis code.google.com/p/pandoc/downloads/ et exécutez-le avec des droits d'administration. L'installation s'effectue automatiquement dans le compte de l'utilisateur courant (dossier `C:\Users\<username>\AppData\Local\Pandoc`). La variable `PATH` de l'utilisateur est complétée de façon que la commande `pandoc` soit accessible depuis n'importe quel répertoire (dans une fenêtre **Invite de commande**). Finalement le menu **Démarrer** de l'utilisateur est complété par un raccourci vers le *Pandoc User's Guide*.

Sous Mac OS X

Téléchargez l'image-disque `pandoc-<version>.dmg` depuis code.google.com/p/pandoc/downloads/. Ouvrez-la, lancez le `pandoc-<version>.pkg` et suivez les indications de cet installateur.

Sous GNU/Linux

La méthode la plus simple consiste à installer la version de Pandoc packagée dans le dépôt officiel de votre distribution Linux. Sous Debian et dérivés ainsi que Fedora, il s'agit du *package* nommé `pandoc` que l'on installe donc de façon habituelle avec: `sudo apt-get install pandoc` (Debian/Ubuntu) ou `yum install pandoc` (Fedora). On obtiendra cependant ainsi des versions de Pandoc datant d'environ une année (1.9.1 sous Ubuntu 12.04 LTS et 1.9.4 sous Fedora 18) dans lesquelles nombre de possibilités présentées ci-dessus ne sont pas encore implémentées.

Pour installer la version la plus récente (1.11.1 au moment de rédiger cet article), étant donné qu'il n'existe actuellement sous Ubuntu pas de PPA (dépôt alternatif de paquets logiciels) relatif à Pandoc, il s'agit donc de l'installer/compiler soi-même en procédant ainsi:

- installation préalable de la plate-forme Haskell (langage de programmation dans lequel est écrit Pandoc) avec: `sudo apt-get install haskell-platform` (une soixantaine de *packages* seront installés occupant ~500 MB);
 - puis installation/compilation proprement dite de Pandoc avec: `cabal update && cabal install pandoc` (ne prenez pas garde aux messages d'avertissement); cette commande pourra être ultérieurement répétée pour bénéficier des mises à jour de Pandoc;
 - l'installation s'effectue alors pour l'utilisateur courant dans le répertoire `~/cabal` (arborescence *standalone* qui pourrait être copiée sur une autre machine), l'exécutable étant `~/cabal/bin/pandoc`;
 - mettez à jour votre variable d'environnement `PATH` en conséquence (en introduisant p.ex. la ligne `export PATH="$HOME/~/cabal/bin:$PATH"` au bas de votre prologue `~/profile`).
- La procédure est similaire sous Fedora.

Quel que soit le système d'exploitation

Pour être en mesure de faire une conversion directe **Markdown** → **PDF**, Pandoc nécessite qu'un environnement **LaTeX** soit présent sur votre machine, par exemple TeX Live sous Linux, MiKTeX sous Windows, ou BasicTeX sous Mac OS X.

Utilisation du convertisseur Pandoc

En premier lieu il est important de dire que Pandoc utilise exclusivement l'encodage de caractères **UTF-8** en entrée et sortie. Si vous obtenez des erreurs lors d'une conversion, le plus vraisemblable est que votre fichier Markdown d'entrée est encodé en ISO-8859-* (ISO-Latin), donc commencez par le re-sauvegarder en UTF-8.

Pandoc est un outil qui s'utilise en ligne de commande, donc *a priori* plutôt dans une fenêtre terminal ! Si cela vous ne vous convient pas, vous pouvez définir, dans votre éditeur de texte ou IDE (environnement de développement intégré), les commandes de conversion utiles [8]. La forme générale de la commande Pandoc est la suivante: `pandoc option(s) fichier(s)_entree` ou `pandoc option(s) URL`. Voici quelques exemples:

```
pandoc -o sortie.out entree1.in entree2.in
convertit le résultat de la concaténation des fichiers entree1.in entree2.in dans le fichier sortie.out. Le format d'entrée est automatiquement déduit de l'extension .in des fichiers d'entrée (p.ex. .md ⇒ Markdown/Pandoc), et le format de sortie de l'extension .out du fichier de sortie (p.ex. .html ⇒ HTML);
```

```
pandoc -f html -t markdown url
convertit en Markdown étendu la page Web correspondant à l'url spécifiée, et envoie le résultat sur la sortie standard;
```

```
pandoc -o sortie.odt entree.md
convertit le fichier Markdown/Pandoc entree.md en un fichier sortie.odt au format ODF (LibreOffice Writer).
```

Parmi les très nombreuses *options* de la commande Pandoc, nous ne présentons ci-après que celles qui nous semblent les plus utiles pour un usage courant. Sachez que:

- toutes les options disponibles sont indiquées par les commandes `pandoc -h` et `pandoc --help`;
- la description détaillée de celles-ci est accessible avec `man pandoc` sous Linux ou Mac OS X, ou dans le *Pandoc User's Guide* sous Windows.

Vous trouvez sous [12] toute une série d'exemples de fichiers et conversions Pandoc, avec les commandes associées.

Options générales

```
-o fichier.ext
définit le nom du fichier de sortie, et déduit le format à partir de l'extension ext spécifiée. Sans cette option, le résultat de la conversion est envoyé sur la sortie standard.
```

```
-f format OU --from format
définit explicitement le format d'entrée (prime sur le format dérivé de l'extension du fichier d'entrée).
```

```
-t format OU --to format
définit explicitement le format de sortie (prime sur le format dérivé de l'extension du fichier de sortie).
```

Tous les *formats* d'entrée et sortie supportés sont indiqués par la commande `pandoc -h`, ou par `man pandoc` sous les options `-f` et `-t`. Le format `markdown` désigne le *Pandoc extended Markdown*, et `markdown_strict` utiliserait l'*original unextended markdown* (ignorant les extensions Pandoc).

```
-s OU --standalone
applique le template par défaut associé au type de fichier de sortie. S'agissant d'une conversion en HTML, ce template (visible par la commande pandoc -D html) ajoute donc notamment le bloc <html><head>...</head> ainsi que les balises <body> et </body></html>. L'option --template=fichier permettrait d'utiliser son propre fichier de template.
```

Conversion en HTML

```
-c fichier.css
associe la feuille de style fichier.css au fichier HTML généré.
```

```
--self-contained
produit un document HTML entièrement standalone, c'est-à-dire sans dépendances externes. Les images sont ainsi incorporées au code HTML, de même que les feuilles de style, les scripts JavaScript, etc. Très utile pour les slide shows!
```

Numérotation des titres, table des matières

```
-N
numérote automatiquement les titres et sous-titres <h*>.
```

```
--toc
incorpore une table des matières des titres. Sont collectés les titres du nombre de niveaux levels spécifié par l'option supplémentaire --toc-depth=levels. En l'absence de cette option, seuls les titres des trois premiers niveaux sont collectés (en HTML: <h1>, <h2>, <h3>). Avec le template par défaut, la table des matières est placée au tout début du document, juste après le bloc de titre.
```

Formules, slide shows

```
-s --latexmathml OU -s --mathml OU -s --mathjax OU
-s --jsmath=script.js
utilise l'une des méthodes spécifiées pour convertir les formules TeX Math en MathML afin de les afficher dans les navigateurs Web supportant ce standard du W3C.
```

```
-s -t format -i --self-contained
produit un slide show HTML au format spécifié (qui peut être de type s5, slidy, slideous ou dzslides). Avec l'option --self-contained, le fichier HTML sera totalement standalone (incorporation des données liées telles que les images, scripts, CSS...). L'option -i fera apparaître les éléments de listes progressivement à chaque clic de souris (sinon le clic passe directement au slide suivant). L'option supplémentaire --slide-level=level_number permet de définir le critère de découpage du document en slides.
```

Exportation en PDF

On a vu que la conversion directe **Markdown** → **PDF** par Pandoc nécessite de disposer d'un environnement LaTeX sur sa machine. Une manière alternative de procéder (sans utiliser LaTeX) consiste à effectuer, en 2 étapes:

- une conversion Markdown → HTML avec Pandoc
- puis une conversion HTML → PDF (associé à une bonne feuille de style CSS) par votre navigateur Web et un driver d'impression PDF, ou directement en mode commande avec l'outil `wkhtmltopdf` mentionné un peu plus bas.

Éditeurs appropriés à l'édition Markdown

Pour faciliter la rédaction et le confort de la lecture, vous avez tout intérêt à utiliser un éditeur de texte supportant la coloration syntaxique Markdown (*syntax highlighting*).

Parmi les éditeurs libres ou gratuits, nous pouvons vous suggérer:

- 🪟 sous Windows: Notepad++ avec le plugin `markdown_npp` (https://github.com/thomsmits/markdown_npp)...
- 🍏 sous Mac OS X: TextWrangler, BBEdit...
- 🐧 sous GNU/Linux: Gedit, Geany, vim...

N'hésitez pas à nous signaler vos trouvailles !

Au sujet de cet article

Parvenus au terme de cet article, nous espérons vous avoir convaincu de la simplicité, l'efficacité et la beauté du balisage léger Markdown ainsi que la puissance de Pandoc ! Vous pouvez ainsi envisager l'utilisation de ces techniques pour élaborer divers documents (là où la puissance de LaTeX n'est pas nécessaire), de la documentation HTML, des *slides shows*, des *e-books*... Sachez que cet article a lui-même été élaboré de cette manière, et que vous pouvez consulter en ligne ses versions:

- source:** enacit1.epfl.ch/markdown-pandoc/article-fi-juillet-2013.md,
- HTML:** enacit1.epfl.ch/markdown-pandoc/article-fi-juillet-2013.html,
- PDF:** enacit1.epfl.ch/markdown-pandoc/article-fi-juillet-2013.pdf,
- ePub:** enacit1.epfl.ch/markdown-pandoc/article-fi-juillet-2013.epub.

Comme il s'agissait de présenter ici les fonctionnalités de Pandoc, nous avons bien entendu largement fait usage des extensions apportées à la syntaxe de base Markdown. Pour parvenir, sans gros efforts, à la mise en page relativement élaborée des versions dérivées HTML et PDF, nous avons en outre eu recours aux technologies HTML suivantes:

- le rendu de la page HTML s'appuie sur une feuille de style CSS de notre propre création: enacit1.epfl.ch/markdown-pandoc/markdown-pandoc.css; la conversion en HTML a été réalisée par la commande `pandoc -s --toc --self-contained --latexmathml -c markdown-pandoc.css -o article-fi-juillet-2013.html article-fi-juillet-2013.md`;
- pour définir des blocs de textes côte à côte, nous avons fait usage, dans le fichier source, de balises HTML `<div>`, technique plus souple (moyennant quelques règles CSS) que de recourir aux Tableaux Pandoc;

- la version PDF a été dérivée de la version HTML (et non pas du code source Markdown/Pandoc) en utilisant l'outil en mode commande `wkhtmltopdf` (**WebKit HTML to PDF**, téléchargeable depuis code.google.com/p/wkhtmltopdf/), héritant ainsi de notre stylage CSS;
- pour la version ePub, nous n'avons apporté que quelques petites adaptations à la feuille de style CSS utilisée pour le rendu HTML, mais le fichier passe ainsi déjà très bien dans la liseuse libre Calibre (calibre-ebook.com).

Concernant le second point: bien que Pandoc permette l'usage de balises HTML dans un fichier Markdown, notez que cette manière de procéder n'est pas recommandée, ces balises étant ignorées en cas de conversion directe par Pandoc dans d'autres formats que HTML ou ePub (par exemple en PDF, ODT, LaTeX...).

Quelques références

- [1] BORBOËN, Nicolas. *Markdown*. In: FI 1/2013. flashinformatique.epfl.ch/spip.php?article2629.
- [2] WIKIPEDIA. *Lightweight markup language*. En ligne. en.wikipedia.org/wiki/Lightweight_markup_languages.
- [3] GALLAIRE, Florent. *Comparaison de langages de balisage légers*. fgallaire.flexnet.com/comparaison-langage-balisage-markup-lightweight-leger-txt2tags-pandoc-docutils-asciidoc-deplate-stx2any-aft-markdown-textile/.
- [4] FABBRI, Pascal. *AsciiDoc pour la production rapide de documents*. In: FI 3/2012. flashinformatique.epfl.ch/spip.php?article2527.
- [5] GRUBER, John. *Markdown Syntax*, sur le site de son auteur. daringfireball.net/projects/markdown/syntax.
- [6] MACFARLANE, John. *Pandoc Project Home*. En ligne. code.google.com/p/pandoc/.
- [7] MACFARLANE, John. *Pandoc User's Guide*. En ligne. johnmacfarlane.net/pandoc/README.html.
- [8] BONJOUR, Jean-Daniel. *Élaboration de documents avec Markdown et Pandoc*. Avril 2013. En ligne. enacit1.epfl.ch/markdown-pandoc/.
- [9] SANSON, David. *Pandoc Cheatsheet*. En ligne. <https://github.com/dsanson/Pandoc.tmbundle/blob/master/Support/doc/cheatsheet.markdown>.
- [10] MACFARLANE, John. *Gitit Demo*. En ligne. gitit.johnmacfarlane.net.
- [11] FORUM, Google Groups. En ligne. groups.google.com/group/pandoc-discuss.
- [12] MACFARLANE, John. *Pandoc Demos*. En ligne. johnmacfarlane.net/pandoc/demos.html.



Article du FI-EPFL 2013 sous licence CC BY-SA 3.0 / J.-D. Bonjour