Pandoc入門：MarkdownからHTML・PDF・Writer/Word文書・スライドを生成する

OSC 京都 2017 1 日目 セミナー

藤原 由来（日本 Pandoc ユーザ会）

2017 年 8 月 4 日

この発表について

- 文書変換ツール「Pandoc」の入門セミナーです
- 対象者
  - ドキュメンテーション（文書の作成・処理）に興味のある方
  - 文書の作成・処理を効率化したい方
  - Markdown などの軽量マークアップ言語をうまく活用したい方
- 前提とする知識：基本的なコマンドラインの使い方
  - ターミナル (Linux/Mac)
  - コマンドプロンプト (Windows)

自己紹介

- 名前
  - 藤原 由来 (本名)
    - GitHub
    - Facebook
  - すかいゆき・藤原 惟
    - Twitter
- 職業
  - フリープログラマ・技術ライター
  - 専門学校 非常勤講師

Pandocに関する活動

- Pandoc ユーザーズガイドを和訳
  - Pandoc ユーザーズガイド 日本語版
  - バージョンが古くなったので、改訂を予定
- Qiita 等に記事執筆
  - 多様なフォーマットに対応！ ドキュメント変換ツール Pandoc を知ろう- Qiita
- 日本 Pandoc ユーザ会
  - 最近 Slack 作りました: Slack 登録フォーム

Pandoc公式サイト

- Pandoc - About pandoc
- ユーザーズガイド
  - Pandoc - Pandoc User’s Guide
  - Pandoc ユーザーズガイド 日本語版

いきなりですが質問です

Q1: 普段はどんなファイルやドキュメントを扱っていますか？

- HTML
- LaTeX (数式の入ったドキュメント)
- XML 系のドキュメント
- プログラミング言語のドキュメント機能
  - Javadoc, Python docstring など
- Microsoft Office
  - Word, Excel, PowerPoint
- LibreOffice (Apache OpenOffice)

Q1: 普段はどんなファイルやドキュメントを扱っていますか？

- 軽量マークアップ言語
  - Markdown
  - reStructuredText (Sphinx)
  - Emacs org-mode
  - Wiki 記法
    - MediaWiki (Wikipedia)
- その他

Q2: どのような目的でドキュメントを扱っていますか？

- Web への公開（CMS・ブログ・Wiki 含む）
- 組織内の情報共有
  - 社内 Wiki
  - プロジェクト管理（ガントチャート・UML も含む）
- 組織外との情報共有・コミュニケーション
- 自分で読み返すためのメモ
- その他

この発表でやること

- Pandoc の概要
- Pandoc をインストールする
- Pandoc の基本的な使い方
  - Markdown ↔ LibreOffice Writer を例に
- Pandoc の応用
- まとめ・お知らせ

この発表でやらないこと

- プログラミング言語のドキュメント機能
  - Pandoc 自体は Haskell 用ドキュメンテーションに対応
    - Haddock, Literate Haskell
  - Pandoc との組み合わせができる場合もある
- 表形式のドキュメント（Excel, CSV など）
  - 現状の Pandoc が扱える文書モデルから離れるので
  - 「文書の一部」として扱うことは Pandoc で可能です

Pandocの概要

こんなことに困っていませんか？

- MS Word/LibreOffice Writer
  - 重いのでテキストファイルで下書きしたい
  - バージョン管理をしたいけど、Word 文書は Git 管理が面倒
- オフィスにある大量の文書を別の書式に変換したい
- MediaWiki 記法で書いた原稿を Sphinx(reST) で使いたい
- Markdown でスライドショーを作りたい

Pandocとは

- 文書変換ツール
  - あるフォーマットで書かれた文書を、別のフォーマットに変換するツール
  - Pandocの特徴は、対応フォーマットが非常に多いこと
- Pandoc公式サイト
  - 「a universal document converter」
  - 汎用ドキュメントコンバータ
- オープンソースソフトウェアの1つ
  - ソースコード: jgm/pandoc
  - ライセンス: GPL2

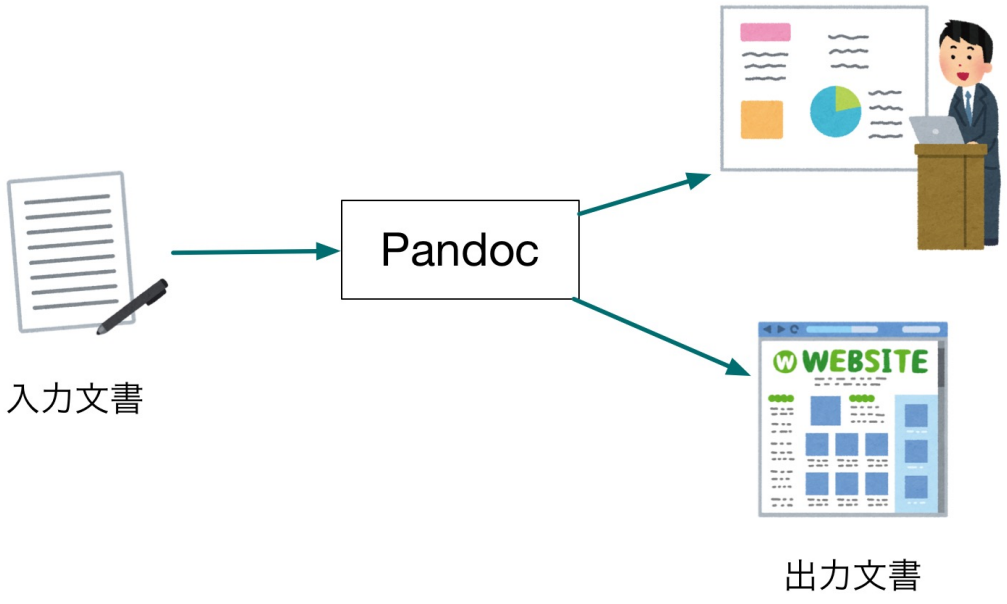
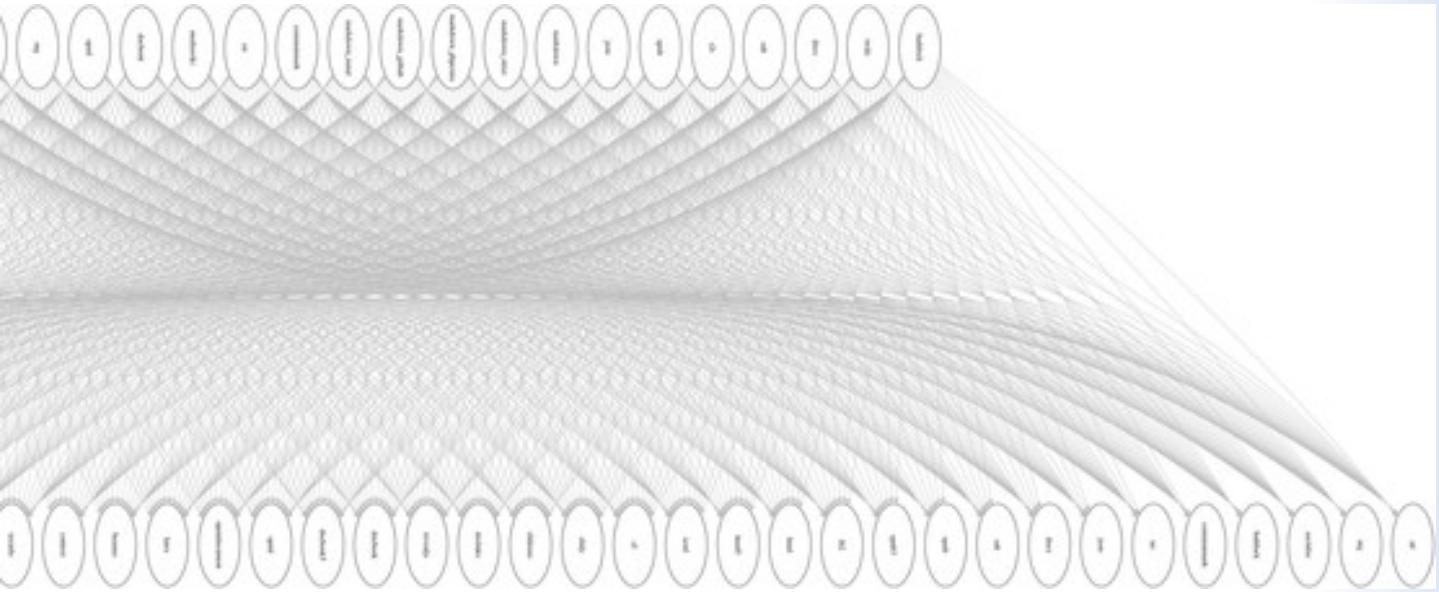


Figure 1: Pandoc の処理フロー

対応フォーマット（一部省略）

- 入力
  - **Markdown** (Pandoc, CommonMark, PHP Markdown Extra, GitHub-Flavored Markdown, MultiMarkdown)
  - (subsets of) Textile, **reStructuredText, HTML, LaTeX, MediaWiki markup**, Emacs Org mode
  - OPML, DocBook, EPUB, **ODT** and Word docx
- 出力
  - 入力フォーマットのほとんど（ODT 含む）
  - **Markdown**
  - man ページ, AsciiDoc, InDesign ICML
  - プレゼンテーション: LaTeX Beamer, HTML5(reveal.js など)
  - PDF (wkhtmltopdf または LaTeX エンジンが必要)



Markdown ?

そもそもMarkdownって何？

- このスライド自体が、実はMarkdownで書かれています
- 元々はJohn Gruberが作ったオリジナルの処理系 でHTMLに変換するための記法だった
- そのうちGitHubやPHPなどで記法が拡張された
  - MultiMarkdownやPandocの登場をきっかけに、目的も「論文」「プレゼンテーション」「電子書籍」など用途が広がった
  - 数々の「方言」がある状態
- 基本のMarkdownだけを覚えれば、大抵は方言が違っても「大まかには」書ける
  - おすすめ早見表: Markdown Reference (CommonMark)
  - 足りない部分は、各ツール・サービスのドキュメントを参照
  - プレビューを行うのが鉄則

裏方としてのPandoc

- 実は裏で Pandoc が動いているケースもいくつかあります
- R: R Markdown
  - RStudio という統合環境の中で使える
  - 厳密には knitr の機能
- Python: Jupyter Notebook
  - nbconvert
- テキストエディタ: Typora
  - Markdown エディタの一つ (Win/Mac/Linux)
  - コマンド苦手な人でも、Pandoc の変換機能を使えます

Pandocでできないこと

- 表主体の文書を扱うこと
  - Excel, LibreOffice Calc
  - 一部に簡単な表を埋め込むことはできる（HTML の`<table>`相当）
- PowerPoint/LibreOffice Impress に変換すること
  - LaTeX Beamer/HTML プレゼンには変換可能

Pandocを使う心得

- 過剰な期待をし過ぎないこと
  - Pandoc は万能でないし、文書仕様の全てを満たしているわけではない
- 補助的に使うのがベスト
  - Pandoc で、テキストと大まかな構造を抽出
  - 変換し切れなかった部分を、手作業や自作スクリプトで編集

Pandocの実装

- 言語: Haskell
  - Pandoc 的には、「厳密に型が定義されている」ことがありがたい
  - Haskell は構文解析器 (パーサ) を作るのにすごく適している (Parsec など)
- モジュール構成
  - Reader: 入力文書を解析し、Haskell 上の中間文書に変換する
  - Writer: 中間文書を受け取り、出力フォーマットに変換する

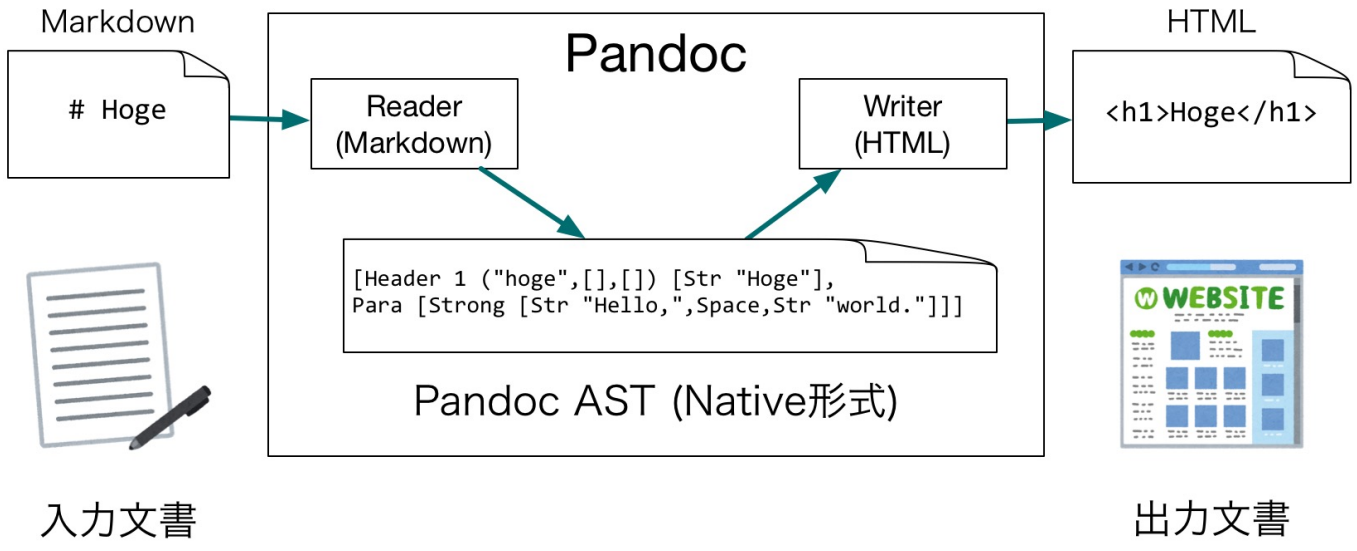


Figure 2: Pandoc の処理フロー（詳細）

Pandocにおける Markdown

Pandocが扱えるMarkdown方言

- Pandoc's Markdown: `-f markdown`
  - Pandocにおける標準のMarkdown方言
  - 技術文書から論文・電子書籍まで幅広く対応
- GitHub Flavored Markdown (gfm): `-f markdown_github`
  - GitHubの標準、プログラマ・フレンドリーな方言
- PHP Markdown Extra: `-f markdown_phpextra`
  - 最近はMarkdown Extraとも呼ばれる
- MultiMarkdown: `-f markdown_mmd`
  - HTMLだけでなくLaTeXなどの論文も意図した処理系
- CommonMark: `-f commonmark`
  - 仕様の曖昧さをなくすことを目的とした仕様/処理系
  - 事実上の標準？（RFCなどによる正式な標準ではない）

Pandoc’s Markdownの特徴

- 詳しくは Pandoc’s Markdown を参照
  - または Pandoc ユーザーズガイド 日本語版 を参照
- HTML の定義リスト (<dl>, <dt>, <dd>) がある
- 表 (<table>) は 4 種類ある
  - 日本語には「Grid Table」か「Pipe Table」がおすすめ
- ヘッダ部分にメタデータを記述できる（重要）
  - タイトルブロック（行を%で始める）
    - タイトル（1 行目）、著者（2 行目）、日付（3 行目）のみ簡潔に書ける
  - YAML メタデータブロック（次のスライド）

YAML メタデータブロック

- ブロックを---で始めて . . . で閉じる
- 改行などの書き方は、YAML の文法に従う
  - 参考: Rubyist Magazine - プログラマーのための YAML 入門 (初級編)
- このブロックで定義されたデータは、メタデータという種類の変数となる
  - メタデータは文書変換する際のオプションや制御に使うことができる
    - `pandoc -D`（出力書式の名前）で、実際の使われ方がわかる
  - このブロックを使えば、文書自体に文書変換のオプションを埋めこめる（便利）

YAMLメタデータブロックの例

```
---
title: Pandoc入門：MarkdownからHTML・PDF・Writer/Word文書・スライドを生成する
author: 藤原 由来
date: 2017年8月4日
revealjs-url: reveal.js-3.4.0
theme: sky-sky-y
transition: fade
transitionSpeed: fast
slideNumber: true
...
```

Pandocをインストールする

ターミナルを開く

- Linux/Mac: ターミナル
- Windows: コマンドプロンプト
  - 分かっている方は、お好きなターミナル・シェルでも OK
- 基本的なコマンド操作については、今回は説明しません
  - コマンドが苦手な方は「何ができるか」を覚えてもらえれば幸いです

Pandocのインストール: インストーラ編

- Windows/Mac の場合
- パッケージを直接落としてインストール
  1. ここからパッケージをダウンロード
     - Windows: `.msi`, Mac: `.pkg`
  2. インストール

Pandocのインストール: パッケージマネージャ編

- Mac(Homebrew)
  - `$ brew install pandoc`
- Windows(Chocolatey)
  - `> cinst -y pandoc`

Pandocのインストール: Linux編

- Linux
  - pandoc/INSTALL.md を参照
  - 各種パッケージマネージャでインストールできます
    - Debian, Ubuntu, Slackware, Arch, Fedora, NiXOS, openSUSE, and gentoo
    - 各々のパッケージマネージャで「pandoc」を search/install してください
    - バージョンが古いことがあるので注意
  - ソースコードからビルド
    - Haskell のソースコードをビルドする必要があります
    - Stack(Haskell ビルドツール) を
    - ソースコード: GitHub - jgm/pandoc

wkhtmltopdfのインストール

- PDF 出力のために必要
  - TeXLive を入れていれば、LaTeX 処理系も利用可能（説明略）
  - ただし、pLaTeX は NG なので、LuaLaTeX/XeLaTeX が必要です
- インストーラを直接落としてインストール
  1. wkhtmltopdf - Downloads からダウンロード
  2. インストール
- パッケージマネージャでインストール
  - Mac(Homebrew): `$ brew cask install wkhtmltopdf`
    - Cask の方なので注意
  - Windows(Chocolatey): `> cinst -y wkhtmltopdf`
  - Linux: 略（各々のパッケージマネージャで「wkhtmltopdf」を search/install してください）

動作確認1: Pandoc単体

※ **藤原の環境：**Windows (Chocolatey) + MSYS2

```
$ pandoc --version
$ pandoc --list-input-formats
$ pandoc --list-output-formats
$ echo "**Hello**" | pandoc -f markdown -t html
<p><strong>Hello</strong></p>
```

動作確認1: Pandoc単体

```
$ echo "**Hello**" | pandoc -f markdown -t html
```

- シェルのパイプ機能を使っています
  - echo が出す標準出力をパイプ (|) で pandoc の標準入力に渡す
  - pandoc は入力・出力ファイル名が与えられてない場合、標準入力・標準出力を使う
- -f: 入力フォーマット名 (from)
  - 使えるフォーマット名は pandoc --list-input-formats で確認できる
- -t: 出力フォーマット名 (to)
  - 使えるフォーマット名は pandoc --list-output-formats で確認できる

動作確認2: ファイルを入力

- 次の内容をテキストファイルで保存し、「hello.md」と保存する

```
# Hello
こんにちは
```

動作確認2: ファイルを入力

- コマンドを実行:

```
$ pandoc hello.md -o hello.html
```

- オプションのない引数 (hello.md): 入力ファイル名
- -o: 出力ファイル名 (output)
  - -t（次スライド）を指定しない場合、拡張子から出力フォーマットを推測してくれる
- 注意: **Pandoc が対応している文字コードは UTF-8 のみです**
  - UTF-8 以外を扱う場合は、nkf/iconv/uconv などの文字コード変換ツールをパイプ (|) に繋げます

動作確認3: Pandoc + wkhtmltopdf (PDF)

```
$ echo "**Hello**" | pandoc -f markdown -t html5 -
o hello.pdf
```

- `-f`: 入力フォーマット名 (from)
- `-t`: 出力フォーマット名 (to)
  - 注意: wkhtmltopdf で PDF を出すときは `-t html5` を指定
    - 内部で文字通り、HTML5 に変換してから PDF に出すので
- `-o`: 出力ファイル名 (output)
  - 注意: wkhtmltopdf で PDF を出すときは、`-o` の拡張子は `.pdf` を指定

Pandocの基本的な使い方

これからやること

- Markdown ↔ LibreOffice Writer の相互変換を例にします
  - 他の書式に変換するときの基礎になります
  - MS Word を扱うときは、ほぼ同じです

これからやること

- Markdown 文書から Writer 文書に変換する
  - とりあえず変換してみる
  - 綺麗な Writer 文書を生成する: reference 機能
- Writer 文書を Markdown などに変換する
- 以下の作業では、GitHub リポジトリの sample ディレクトリにあるファイルを使います
- atarashii_kenpo.md: あたらしい憲法のはなし (Markdown 版) より
  - nogajun さん編、Public Domain

とりあえず変換してみる: pandocコマンド

```
$ pandoc atarashii_kempo.md -o atarashii_kempo.odt
```

- `-o`: 出力ファイル名
  - 多くの場合拡張子で判断できる

ファイルを開く

```
$ open atarashii_kempo.odt       # Mac
$ xdg-open atarashii_kempo.odt   # Linux (GNOME, KDE, Xfce)
> start atarashii_kempo.odt      # Windows
```

綺麗なWriter文書を生成する

- Pandoc の reference 機能を使う
  - ❶ Pandoc コマンドから reference ファイルを生成
  - ❷ reference.odt を Writer で編集する
  - ❸ Pandoc の変換時に reference.odt をオプションで指定する
    - もしくはユーザデータディレクトリに入れる
- Word の場合は、「odt」を「docx」に読み替えると同様にできます

(1) Pandocコマンドからreferenceファイルを生成

```
$ pandoc --print-default-data-file reference.odt > referenc
```

- コマンドオプションで指定する場合は「reference.odt」という名前でなくてよい
- ユーザデータディレクトリ (後述) に入れる場合は必ず「reference.odt」という名前にする

(2) reference.odt を Writer で編集する

- reference.odt を LibreOffice で開く
- reference.odt の内容は Pandoc から参照されない
  - デフォルトで「Hello World!」と表示されている部分のこと
  - 例えば表示用サンプルを置いてもいい
- 「スタイルと書式設定」から書式を変更

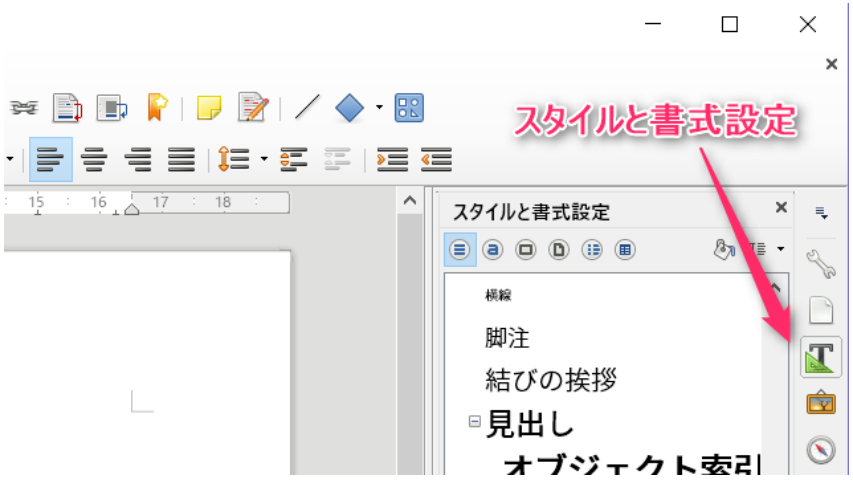


Figure 3:

(3) Pandocの変換時にテンプレートをオプションで指定する

注意: バージョンによって使用するオプションが違います

- `--reference-odt`: Pandoc 2.xの指定
- `--reference-doc`: Pandoc 1.xの指定
- バージョンは `pandoc -v` で分かります
- 実際に使えるコマンドは `pandoc -h` で分かります
  - UNIX系なら `pandoc -h | grep 'reference'` で絞れるはず

```
$ pandoc atarashii_kenpo.md --reference-odt=reference.odt -
o atarashii_kenpo.odt
$ pandoc atarashii_kenpo.md --reference-doc=reference.odt -
o atarashii_kenpo.odt
```

補足: reference.odtをユーザデータディレクトリに入れる

- ユーザデータディレクトリの場所: `pandoc -v` で出る
  - Windows: `C:\Users\ユーザ名\AppData\Roaming\pandoc`
  - Mac: `$HOME/.pandoc`
- このディレクトリに「reference.odt」という名前でテンプレートを入れると、次回からデフォルトで使ってくれる

具体的なノウハウ

- nogajun さん: Pandoc と LibreOffice Writer で iD エディタのマニュアルを製本する, どうしてこうなった - Days of Speed(2014-12-06)
  - nogajun/pandoc-writer の `pandoc-writer.odt` がテンプレートとして使える
- いくやさん: Pandoc で Markdown を ODT に変換する- いくやの斬鉄日記
  - 画像のサイズを整える (ImageMagick の `mogrify` コマンド)
  - 画像の DPI を変更する (同上)

Writer文書からMarkdown/reST文書に変換してみる

- nogajun さんの `pandoc-writer.odt` を変換してみる
  - nogajun/pandoc-writer (CC BY-SA 4.0)
- Markdown (Pandoc's)
  - `$ pandoc pandoc-writer.odt -o pandoc-writer.md`
- reStructuredText (Sphinx などで使用)
  - `$ pandoc pandoc-writer.odt -o pandoc-writer.rst`

Writer文書からLaTeX文書に変換してみる

- LaTeX
  - デフォルトは LuaLaTeX/XeLaTeX が必要なので注意
    - LuaLaTeX
    - XeLaTeX を使う場合: BXjscls がまた新しくなった（v1.1a） - マクロツイーター
  - `$ pandoc -s pandoc-writer.odt -o pandoc-writer.tex`
  - `-s`: 文書として完全になるようにヘッダ・フッタを付ける (standalone モード)

Q&A

Pandocの応用

Pandocの応用

- オフィスにある大量の文書を別の書式に変換したい
- Markdown でスライドショーを作りたい
- フィルタ機能
- おまけ：電子書籍について

オフィスにある大量の文書を別の書式に変換したい

処理したいファイルが大量にある場合は、スクリプトに Pandoc を組み込みます。

1. 下準備: 他のツールなどで、なんとかして Pandoc が処理できる書式に変換する
   - おすすめ: HTML（多くのツールでエクスポートできるので）
2. Pandoc をスクリプトの中で使う
   - シェルスクリプトで直接使う
   - スクリプト言語の外部コマンド機能で呼ぶ
   - スクリプト言語のライブラリから呼ぶ（古い場合があるので注意）

Pandocをスクリプトとして呼ぶ例（その他）

- 各種エディタのプラグイン・拡張で対応
  - Atom, VS Code, ...
- 特に Vim の場合
  - LaTeX 文書を書くときの補助として「Markdown を書いてその場で LaTeX に変換する」拡張がある
  - TeX で書くのめんどくさい部分は markdown で書いちゃえば最強じゃないかな?【Vim + pandoc】 - Qiita

Markdownでスライドショーを作りたい

このスライド自体をPandocで生成しました

- Pandoc's Markdownの書式に従って原稿を書く
  - もちろんこれ以外の書式でも、Pandocが対応していれば書けます
- Pandocで変換する
  - 今回は「reveal.js」形式（HTML+JavaScriptによるプレゼン）に変換
  - `$ pandoc input.p.md -s -f markdown -t revealjs -o index.html`
    - 実際のファイルは次スライドで
    - `-s`: standalone (ヘッダ・フッタの付いた完全な文書を出力)
- アップロードする
  - GitHub Pagesを使うと、直接GitHubにpushすればアップロードできます
    - この場合は、`.nojekyll`という空ファイルを置かないと、404エラーになるので注意
- その他LaTeX Beamerにも変換できます

実際のソースコード

- このスライド自体が GitHub Pages にアップロードされています
- GitHub リポジトリ
  - 発表用スライド (HTML/reveal.js)
  - スライドの Markdown ソース (Pandoc’s Markdown)
  - 復習用資料 (GitHub Flavored Markdown)
  - 変換の補助に Gulp を使っています（`make` のようなもの）

フィルタ機能

- 中間文書を JSON 形式に出す
- それを外部スクリプトが標準入力で受け取り処理する
- それを標準出力に出して、Pandoc に戻す

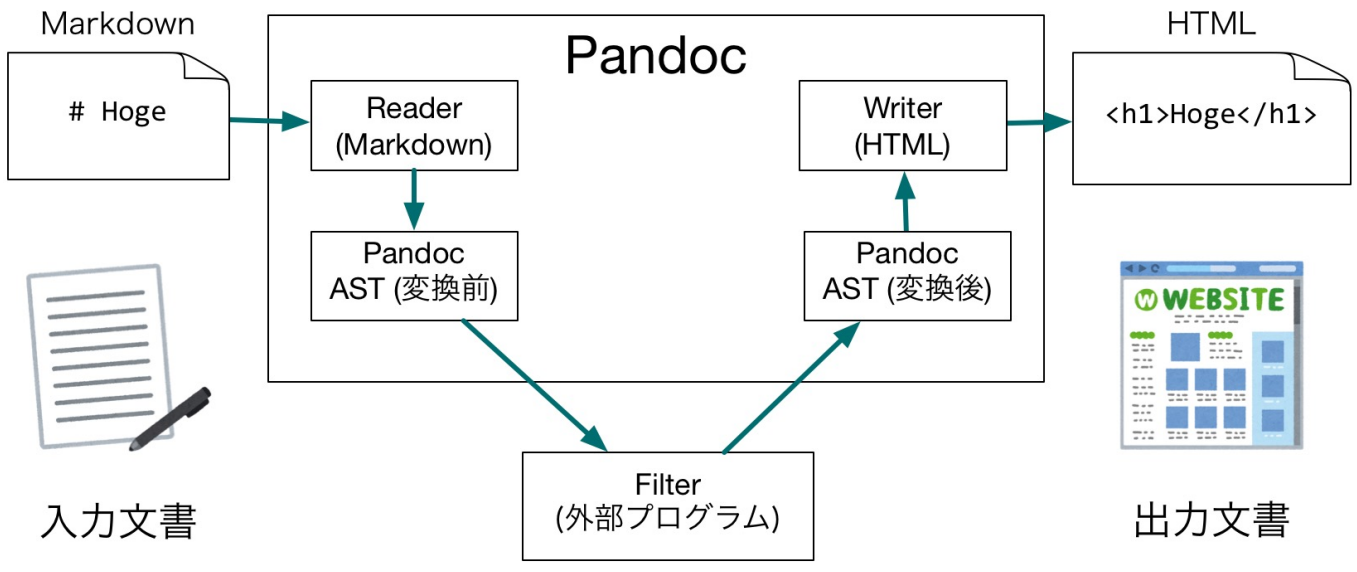


Figure 4: Pandoc の処理フロー（フィルタ付き）

フィルタ機能ができること

- 詳しくは Pandoc Filters (GitHub Wiki) を参照
- 引用を入れる (pandoc-citeproc, pandoc-crossref)
  - Markdown と Pandoc を使って論文っぽい文章を書く | inoblog
- 図表を入れる
  - (mermaid-filter): mermaid で使えるフローチャート、ガントチャート、シーケンス図
  - pantable: CSV ファイルを表として読み込む
- 外部ファイルの読み込み
  - Node.js で書くチュートリアル: pandoc で Markdown を拡張しコードをインポート出来る filter を書く | Web Scratch
- 過去のチュートリアル「Haskell で Pandoc フィルタを実装する」
  - Haskell with Skype Pandoc チュートリアル 第 2 回

おまけ：電子書籍について

- Pandoc も EPUB 出力できる
  - 素朴な EPUB なら日本語でも`-t epub3` で出力できる
- Markdown → EPUB 変換には「でんでんコンバーター」をおすすめします
  - ルビや縦中横が使えて、細かい設定や組版がしやすい
  - 記法: でんでんマークダウン
    - PHP Markdown Extra ベース
- 提案
  - Pandoc に対応する好きな記法で原稿を書く
  - `pandoc -t markdown_phpextra` で、でんでんマークダウン向けに変換
  - でんでんエディターにペーストして仕上げる

まとめ・お知らせ

今日やったこと

- Pandoc の概要
- Pandoc をインストールする
- Pandoc の基本的な使い方
  - Markdown ↔ LibreOffice Writer を例に
- Pandoc の応用

Pandocの今後の課題

- 日本語に特化した文書フォーマットにほとんど対応していない
  - 書籍におけるルビや圏点など
  - 日本語コミュニティの必要性
- 表形式の文書は対応していない
  - Excel 文書など→ Excel 方眼紙への対策には致命的
  - サードパーティのプリプロセッサにより部分的に変換する手段はある
    - 一部の図表（Graphviz など）はこの方法で取り込むことができる
  - 参考: Excel 方眼紙公開討論会 (9/30 @東京)

日本Pandocユーザ会

- Web サイト (リニューアル予定)
  - http://sky-y.github.io/site-pandoc-jp/
- Slack を始めました（どなたでも歓迎します！）
  - Slack 登録フォーム

ドキュメンテーションWiki

- ドキュメンテーション Wiki (GitHub)
- 誰でも編集歓迎します（要 GitHub アカウント）

Q&A

- 連絡先
  - メールフォーム: https://goo.gl/forms/FPB22jv9S5NP4fpx2
  - Twitter: すかいゆき・藤原 惟@sky_y
  - Facebook: https://fb.me/sky.yuki.f