

Mode d'emploi d'Obsidian

1. Pourquoi utiliser Obsidian pour prendre des notes ?

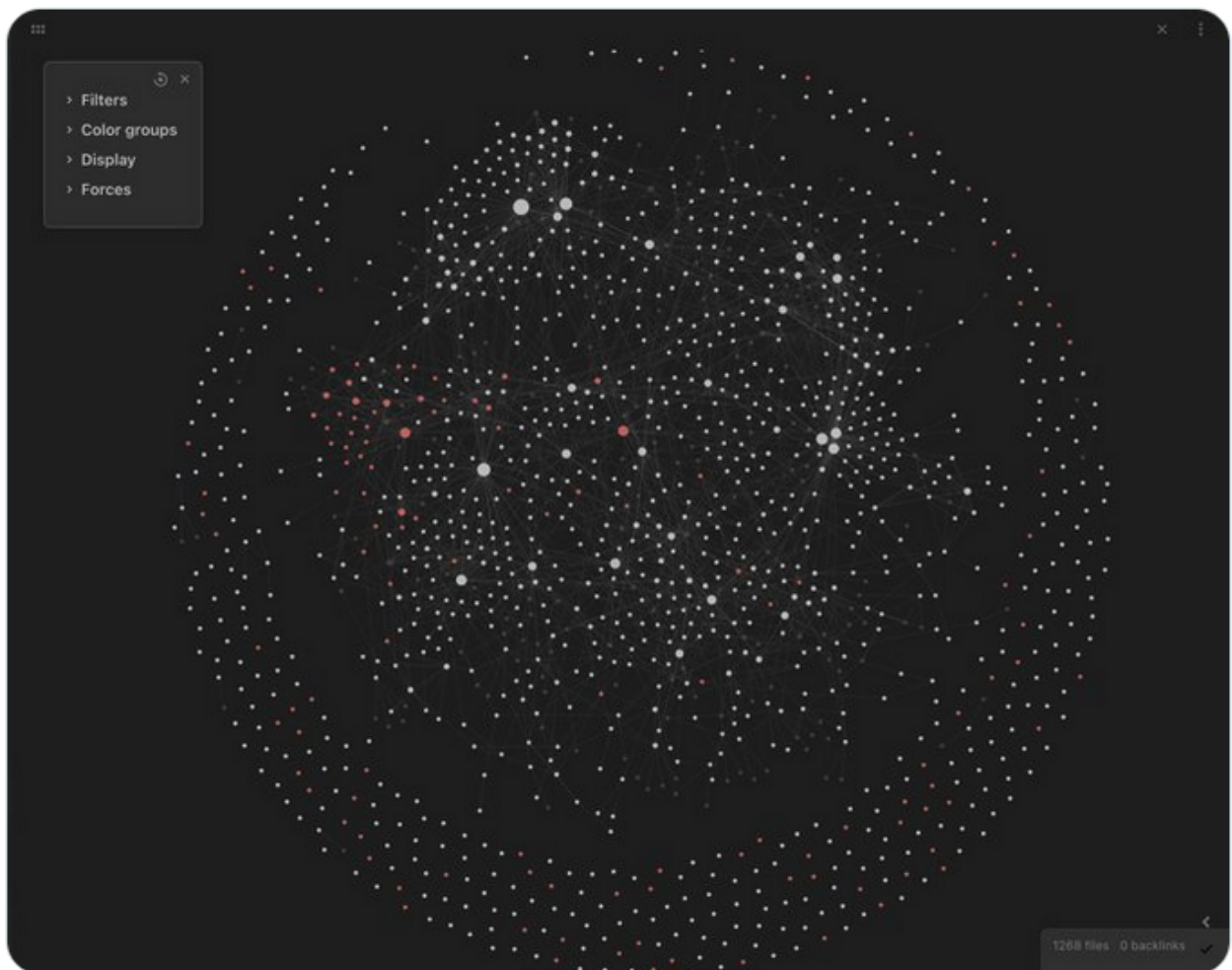


Jan Schaller
@_janschaller



.@obsdmd just keeps on giving. v0.11 introduces (i.a.) persistent folding (💜) and the possibility to color subgroups in your graph! Below: my current graph view, red is everything connected to my PhD! #Obsidian

[Traduire le Tweet](#)



9:45 PM · 16 févr. 2021 · Twitter Web App

1.1 intérêts de la prise de note pour le travail intellectuel

une application de prise de notes doit être jugée selon la manière dont elle remplit les fonctions propres à la prise de notes et qui sont comme le rappelle Baldur Bjarnasson la créativité, la connaissance et la compréhension (*creativity, knowledge, understanding*)[@bjarnasonDifferentKindsNotes20220621](#) :

fonction	fonction détaillée
créativité	engendrer des idées structurées avec des outils heuristiques
créativité	Préserver les idées
créativité	explorer ces idées jusqu'à ce qu'elles soient prises dans un plan cohérent ou bien qu'elles aient permis de répondre à un problème
connaissance	étendre la mémoire : données du client, notes de réunions, références bibliographiques
connaissance	connecter ces données au projet en cours de telle sorte qu'elles soient facilement mobilisables
compréhension	distinguer, recadrer et contextualiser de l'information et des idées pour qu'elles s'intègrent à votre pensée
compréhension	inciter à la reformulation avec ses propres mots des idées prises ailleurs

Obsidian n'est pas vraiment un second cerveau (cette phrase marketing [n'a guère de sens appliquée à une application](#)), mais il nous fournit un excellent environnement de travail pour favoriser ces trois fonctions (créativité, connaissance, compréhension).

Le fait de prendre des notes quotidiennes et de les lier entre elles avec une application de prise de notes en général n'est pas une activité de nature à élever notre quotient intellectuel, c'est une pratique qui nous permet de produire de meilleurs écrits. La prise de notes appliquée à des problématiques quotidiennes (voyages, cuisine, bricolage), permet également d'augmenter la qualité des produits finis en rendant l'information plus disponible. Mais la prise de notes ne devrait pas constituer une fin en soi, c'est une pratique tournée vers un résultat publiable (ou montrable à autrui)[@bjarnasonGettingBackNotetaking2022](#).

mémorisation, appropriation

Le monde du web est celui de la réplique et du copier-coller. Pour autant copier-coller un texte lu est une facilité qui nous empêche de nous approprier son contenu.

Plutôt que de copier-coller un texte qui nous intéresse dans un outil comme Obsidian, il est nettement préférable de le reformuler avec ses propres mots, comme si on voulait l'expliquer à une personne qui ne dispose pas de la source. On s'assure ainsi qu'on a bien compris soi-même le sens du texte en reliant la nouveauté que constitue pour nous l'énoncé du texte avec ce qu'on sait déjà et avec notre contexte d'apprentissage.

Le passage par l'écriture facilite non seulement l'apprentissage mais aussi la réflexion et la conscientisation, y compris de ses émotions ([voir cette discussion](#) sur reddit).

Il est donc important de restreindre les citations aux passages où l'auteur résume sa pensée de manière particulièrement marquante.

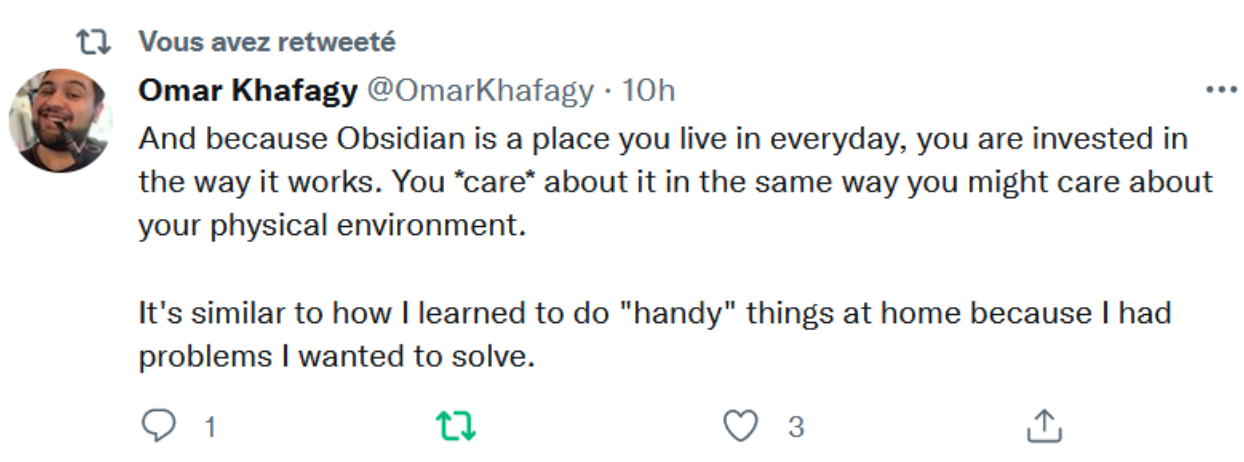
En reliant entre elles ces notes prises au fil de l'eau à la faveur de lectures, Obsidian permet de structurer sa pensée et peut éventuellement amener à des rapprochements inédits entre des concepts qui sont en relation au départ avec des contextes différents (valeur heuristique de ces graphes de notes), c'est l'"émergence informationnelle (information surfacing)" que l'outil peut apporter -un concept en lien avec la [sérendipité](#)- (le concept est d'[Arthur Perret@perretEmergenceInformationnelleInformation2022](#))

L'information qui va permettre d'être productif arrive à la conscience sans avoir été sollicitée.

productivité

Obsidian est un outil qui favorise le *jardinage* des notes [@CollignonIntegratinghypothesisishighlights2021](#). En écrivant une note, on se rend compte qu'on peut faire un lien depuis cette note vers une note plus ancienne. Lorsque le lien est fait, on s'aperçoit qu'on peut ajouter un autre aspect à la note plus ancienne avec le savoir supplémentaire obtenu au moment de l'écriture de la première note. Cette activité aide non seulement à mémoriser, mais à créer de nouveaux liens, à appuyer les connaissances nouvelles sur un socle de connaissances anciennes et à enrichir ces connaissances plus anciennes de nouveaux aperçus.

Obsidian devient rapidement un outil personnel incontournable et nous développons au fur et à mesure que notre usage se déploie de nouvelles compétences pour le maintenir et le rendre encore plus productif



Certains utilisent Obsidian pour se lancer dans l'apprentissage du code par essai et erreur (puisque une note peut embarquer du code) [@KhafagyTwitterPublish2022](#)

1.2 points forts de l'outil

Obsidian présente les avantages de la nouvelle génération d'outils de PKM (personal knowledge management / Gestion personnelle des connaissances)

- des notes liées les unes aux autres par des liens **et des rétroliens** (backlinks)

- des fichiers de notes pouvant contenir d'autres fichiers (par exemple des PDF ou des scripts) selon le principe de la [transclusion](#)
- une structure par défaut horizontale (les notes n'ont pas besoin d'être rangées dans des fichiers séparés cf. [Zettelkasten](#)). C'est possible de le faire si on le souhaite dans Obsidian, cela dit.
- des graphes de notes (cela plaît toujours, même si l'utilité réelle de ces graphes est un sujet débattu depuis que ces outils existent, voir [la partie consacrée aux graphes](#))
- des modèles de note peuvent être produits par l'utilisateur et partagés avec d'autres.
- hébergement en local (pas nécessaire de souscrire à un compte contrairement à Notion ou à Roam, ce dernier étant en plus payant). Tous vos fichiers - plugins, css, images, sons, fichiers en markdown ou en PDF- sont dans un coffre (*Vault*, dans le langage d'Obsidian) que vous hébergez sur votre ordinateur et que vous pouvez synchroniser avec un serveur distant (Github par exemple). Dans ce tutoriel nous traduirons systématiquement "Vault" par "coffre".
- format ouvert des notes ([markdown](#) = format universel, contrairement à Evernote)
- les notes sont hébergées sur notre ordinateur : on en a la maîtrise. Par ailleurs, s'il manque une fonction à Obsidian, il est possible d'y suppléer en utilisant un autre logiciel qui comporte cette fonction et en l'appliquant au coffre-fort (comme dans le cas de [la liaison, déliaison des occurrences](#))
- Pas dépendant d'un éditeur (contrairement à [Org-roam](#) pour Emacs). On peut éditer depuis Obsidian ou bien depuis n'importe quel éditeur qui supporte le markdown (emacs, atom, panwriter, etc.)
- version disponible sur les différentes familles de systèmes d'exploitation (GNU/Linux, Windows, Mac)
- Nombreuses possibilités de recherche dans les notes (par fichier, par mot dans la note, par mot dans la section de la note, par tag...) (voir [@CaseObsidiananswernotetaking2021](#))
- Importance des rétroliens (combien de notes citent une même note). Cela permet de faire des associations d'idées et des graphes
- possibilité d'avoir dans un même endroit des notes et des fichiers divers (code, images, vidéos) et de pouvoir lier ces notes et ces fichiers entre eux.
- le nombre de plugins avec lesquels on peut étendre les capacités d'obsidian est en croissance constante.

1.3 limites

- Obsidian est un logiciel propriétaire, même si la communauté des utilisateurs développe un grand nombre de plugins dont le code est en open source. Si Obsidian décide demain de changer les modes d'accès à son logiciel (par exemple en en faisant un service accessible sur abonnement comme pour Roam, les utilisateurs devront trouver un autre outil libre dont le fonctionnement sera compatible avec le format de leurs notes. Si on veut se tourner vers une alternative entièrement libre comparable à

Obsidian, on peut se tourner vers un logiciel de prise de notes comme Zettlr et pour obtenir un graphe de ses notes prises sur Zettlr, utiliser le logiciel [Cosma](#) mis au point par Arthur Perret (actuellement disponible sur Mac et Windows et bientôt sur Linux).

- Obsidian ne donne pas d'identifiant unique et pérenne à chaque note et ne lie pas ces notes entre elles au moyen de ces identifiants. Selon Arthur Perret, les identifiants uniques et pérennes sont un gage de durabilité. Pour lui, les notes doivent distinguer l'id (identifiant unique), le titre de la note et le nom du fichier. Ce n'est pas le cas avec Obsidian. La méthode Zettelkasten, par ailleurs, utilise par définition ces identifiants uniques et pérennes. Si on se sert de préfixeur Zettelkasten comme identifiant unique, les liens entre fiches ne se baseront pas sur cet identifiant. Plusieurs experts considèrent que c'est une fragilité dans la mesure où les titres et les noms de fichiers peuvent changer, pas les identifiants pérennes. Une solution qui ne comporte pas d'identifiant pérenne par note ne peut pas assurer l'utilisateur que celui-ci pourra réutiliser son système de notes dans la durée, ou bien appliquer une autre solution à son système de notes [@perretLiensWikiIdentifiants2022](#).
- l'affichage des images **dans** l'application n'est pas encore au point. Si la conversion avec pandoc fonctionne sur la taille des images, à l'intérieur d'Obsidian la taille de celles-ci est difficilement modulable.
- Obsidian ne dispose pas de répéteur (spaced repetition), c'est à dire qu'on ne peut pas programmer à l'avance l'apparition de notes à mémoriser (une fonction aléatoire d'affichage de notes existe pourtant sous la forme d'un plugin coeur). Pour l'apprentissage et la répétition, Obsidian peut cependant être connecté à Anki au moyen d'un plugin communautaire.

1.4 Obsidian sur son smartphone

Depuis juillet 2021, Obsidian est disponible sur iOS et Android ([source](#))

Afin de synchroniser ses fichiers avec ceux que contient son ordinateur,

1. télécharger l'app
2. télécharger (si vous n'en disposez pas déjà) d'une app destinée à synchroniser vos fichiers locaux sur smartphone avec ceux du cloud dont vous disposez (par exemple dropsync pour dropbox, foldersync, autosync pour Google Drive ou [Onesync](#) pour Onedrive)
3. connectez votre app de synchronisation avec le dossier dans le cloud que vous utilisez qui contient vos notes
4. synchronisez
5. ouvrez l'app obsidian et choisissez le dossier synchronisé comme coffre.

1.5 une interface agréable

Obsidian offre une interface agréable à l'utilisateur. On peut comme dans Emacs ouvrir plusieurs buffers (fenêtres en même temps) et les organiser.

Certaines curiosités existent pourtant qui peuvent être déroutantes pour un débutant. Par exemple pour zoomer et dézoomer sur une note (sur Windows, Ctrl+ permet de zoomer, c'est classique, **Ctrl 0 permet de revenir à la taille par défaut**)

Les fichiers se trouvent par défaut sur sa machine

Sous Windows, les fichiers de l'application se trouvent sous AppData
(pour accéder à Appdata, afficher les fichiers masqués depuis l'explorateur Windows)

Chaque coffre-fort peut-être hébergé à part. Les coffres se trouvent au même endroit que les plugins (là où on décide de les mettre : cela peut être le disque principal de l'utilisateur, une clé USB ou bien un serveur distant)

Toutes les notes qu'on écrit se trouvent à l'endroit que l'on détermine et on garde la pleine maîtrise de ces notes.

2. La méthode de prise de notes Zettelkasten

Voir [zettelkasten \(méthode\)](#).

3. Rédiger ses notes en [markdown](#)

3.1 Qu'est-ce que Markdown ?

une syntaxe très représentée dans l'écriture web

Markdown (avec une majuscule) est une syntaxe mise au point par John Gruber en 2004 et qui devait simplifier l'écriture web. Les développeurs l'utilisaient (et l'utilisent toujours) pour rédiger plus simplement du HTML en ayant recours à une syntaxe simplifiée par rapport au balisage du HTML.

L'usage du Markdown a fini par se répandre au delà du cercle des développeurs web, et des auteurs, notamment académiques, s'en sont emparés pour écrire des textes convertibles non seulement en HTML mais également dans d'autres formats comme LaTeX ou PDF.

Markdown permet d'utiliser des éditeurs de texte simple compatibles avec un nombre grandissant d'applications (Obsidian en fait partie). L'écriture en texte simple (plain text) a pour intérêt de pouvoir mêler du texte et du code et de se passer de formats propriétaires (par exemple word).

différentes saveurs de Markdown

John Gruber a fait en sorte de ne pas limiter l'usage de sa syntaxe, ce qui fait que beaucoup d'éditeurs se la sont appropriée pour monter leur propre éditeur Markdown. C'est pour cela que d'un outil à l'autre, on peut être confronté à des saveurs (flavors) différentes de

Markdown. Par exemple, un lien ne se formate par exactement de la même manière sur Slackmarkdown et sur Atom.

Le principe de Gruber était d'inciter l'utilisateur à avoir recours directement au HTML chaque fois que rien n'était prévu dans Markdown pour éditer tel ou tel élément du texte (par exemple les notes de bas de page ou les tableaux):

For any markup that is not covered by Markdown's syntax, you simply use HTML itself. There's no need to preface it or delimit it to indicate that you're switching from Markdown to HTML; you just use the tags.

([John Gruber](#))

Pourtant, comme le rappelle Knut Malvaer, les éditeurs ont eu tendance à adapter Markdown et à augmenter ses capacités pour satisfaire leurs usagers parmi lesquels la proportion des auteurs non développeurs n'a cessé de croître, ce qui fait que la syntaxe Markdown n'est plus entièrement unifiée d'un éditeur à l'autre.[@MelvaerThoughtsMarkdown2022](#).

[Commonmark](#) a été créé pour fixer une syntaxe en markdown minimale, transposable d'un outil à l'autre, ce qui aide dans une certaine mesure les développeurs à gérer cette hétérogénéité dans le formatage en markdown (car le markdown est devenu depuis 2004 la *lingua franca* de l'écriture web). Cela n'empêche toujours pas un éditeur d'ajouter les fonctionnalités qu'il souhaite à cette base que constitue Commonmark. Ainsi Rmarkdown continue d'avoir une syntaxe markdown un peu différente de github sur des fonctions spécifiques même si à la base ces deux services utilisent Commonmark.

3.2 niveaux de titre, listes à puces

niveaux de titre

Le titre de la note est distinct du nom du fichier. Le nom du fichier est la première chose qui est créée de la note.

Dans une note, entourer un mot avec un double crochet (`[[mot]]`) crée automatiquement une note comportant `mot.md` dans le coffre.

Une option des paramètres intitulée "toujours mettre à jour les liens internes" permet de maintenir le lien entre la note 1 comportant `[[mot]]` et la note rebaptisée par exemple `word.md`

Si on configure un titre dans le pavé Yaml, celui-ci apparaîtra comme titre du document exporté (par exemple en PDF). Voir la partie concernant [YAML](#)

MD	niveau de titre	HTML
# titre 1	titre de niveau 1	<h1>
## titre 2	titre de niveau 2	<h2>

MD	niveau de titre	HTML
### titre 3	titre de niveau 3	<h3>
#### titre 4	titre de niveau 4	<h4>

numérotation automatique des titres

Il est possible de numéroté automatiquement les entêtes de paragraphe de la manière suivante :

- installer et activer le plugin communautaire *number heading plugins* pour Obsidian
- définir les raccourcis clavier nécessaires pour commander la numérotation des entêtes (ou la supprimer)

liste à puces

le tiret (-) en début de phrase permet de faire une liste à puces :

- pommes
- poires
- bananes

permet d'éditer

- pommes
- poires
- bananes

Attention : la conversion de la note en markdown vers un autre format (ODT ou PDF notamment) peut mettre à mal les listes à puces si chaque élément qui les constitue n'est pas suivi de deux espaces en fin de ligne.

sous liste à puces

- pommes
- golden
- pink lady

Pour éviter d'obtenir :

- pommes - poires - bananes

dans le texte convertir en open document ou en PDF, ajouter systématiquement deux espaces dans le texte en markdown après pommes, poires et bananes.

3.3 tableaux, indentation, épigraphes

éditer un tableau

Code :

```
| colonne A | colonne B | colonne C |  
|:---:|:---|:---:|  
|centre|aligné gauche |aligné droite
```

résultat :

colonne A	colonne B	colonne C
centre	aligné gauche	aligné droite

Pour fusionner des cellules, il est actuellement nécessaire de recourir au HTML, car le markdown ne gère que des cellules équivalentes en taille (voir [Mode d'emploi d'Obsidian > utilisation du HTML > pour faire des tableaux avec des colonnes ou des lignes fusionnées](#))

```
|           |           Grouping           ||  
First Header | Second Header | Third Header |  
----- | :-----: | -----: |  
Content     |      *Long Cell*      ||  
Content     |  **Cell**  |      Cell  |  
section    |  More      |      Data  |  
And more    | With an escaped '\|'      ||  
[Prototype table]
```

Si on dispose d'un tableau en format excel ou calc et qu'on veut le transposer en format markdown, on peut utiliser des ressources en ligne pour cela comme [Table to Markdown](#)

Le *pipe* (|) qui sert à faire des tableaux peut aussi permettre de prendre en compte les espaces en début de phrase (pour une conversion en pdf avec Pandoc)

code :

```
| Crépuscule
|   La nature est en extase de crépuscule
| Et son ode,
|           mon sang,
| Un océan brûlant s'enroule, recommençant ses vagues
|           jusqu'à moi
|
|           (Adonis, *Ismaël*)
```

résultat :

```
| Crépuscule
| La nature est en extase de crépuscule
| Et son ode,
| mon sang,
| Un océan brûlant s'enroule, recommençant ses vagues
| jusqu'à moi
|
| (Adonis, Ismaël)
```

3.3 caractères

MD	niveau de titre	HTML
*	<i>italiques</i>	<i>
**	gras	
~~	barré	<s>
`	code	<code>
==	surlignage	<mark>

Pour "échapper" un caractère qui fait partie de la syntaxe du markdown (*, #, ~ par exemple), il faut faire précéder le caractère en question d'un antislash (\)

3.4 Liens, images, tags

liens internes

Comme on l'a vu plus haut, pour faire un lien interne vers une note existante ou bien en créer une, il suffit de mettre entre double-crochets le mot ou l'expression qui correspond au nom de la note en question.

Si l'on veut pointer vers une section de cette note, rajouter un dièse et sélectionner la partie de la note vers laquelle on veut pointer.

Voici comment se formate par exemple un lien direct vers la partie de ce guide correspondant aux [rétroliens](#) :

Voici comment se formate par exemple un lien direct vers la partie de ce guide correspondant aux **[[Mode d'emploi d'Obsidian#Les rétroliens|rétroliens]]**

syntaxe du lien :

[[nom de la note#nom du header|texte du lien]]

Si on veut lier sur un bloc plutôt qu'un header, remplacer le # par un ^

notes de bas de page

[^1] permet de créer une note de bas de page portant le numéro 1. Exemple:
première note de bas de page^[1]

transclusion

Avec Obsidian, on a la possibilité de faire un lien vers un document de la base de notes (une image, un PDF) de telle sorte qu'en mode lecture, si on passe la souris sur le lien, une prévisualisation du document s'affiche (pour y accéder, il suffit de cliquer) ou bien on peut aussi inclure le document dans la note, de telle sorte que celui-ci soit visible et déroulable. On peut le faire notamment pour un PDF :

![[document.pdf]]

pour afficher une version ancienne de ce guides en PDF à l'intérieur de la présente note :

![[mode_emploi_obsidian.pdf]]

liens externes

[bibliothèque](<https://bibliotheques.univ-rennes1.fr>)

[bibliothèque](#)

![[bibliothèque](https://bibliotheques.univ-rennes1.fr/sites/bibliotheques.univ-rennes1.fr/files/styles/max_2600x2600/public/medias/images/Sante-web.jpg?itok=S0Ax1GSa)

ou

![[bibliothèque](images/BUVS.jpg)



possibilité, comme pour un PDF de faire un glisser déposer du document dans la note.

les tags

Avec les rétroliens, les marqueurs sont une grande richesse de ces outils de gestion de notes.

Comme ailleurs, le tag peut apparaître dans le texte précédé (sans espace) d'un dièse ou hashtag (#) ou bien dans l'entête YAML après (ce qui a l'avantage de ne pas les laisser apparaître dans le document quand celui-ci est exporté dans un format de publication comme wiki, html, pdf, etc.) :

```
tags: [tag1, tag2, tag3/tag4]
```

ici le tag4 est un tag fils de tag3. On peut donc prévoir dans Obsidian, une arborescence de tags.

Attention si on laisse un espace entre le dièse et le mot-clé qui constitue le tag, Obsidian va l'interpréter comme un titre 1 selon la syntaxe Markdown.

Il ne peut y avoir d'espaces dans les tags :

```
tags: [sciences politiques]
```

va générer deux tags : #sciences et #politiques

il faut veiller à prévoir des liaisons entre les mots constitutifs d'un tag, par exemple

l'underscore :

```
tags: [sciences_politiques]
```

On peut mener une recherche sur les tags uniquement.

Dans les plugins standards, il est intéressant d'autoriser l'apparition du volet des tags (tag pane) à droite. Les tags y apparaissent par défaut rangés par ordre de fréquence d'apparition dans les notes)

Si on charge et active le plugin communautaire [Tag Wrangler](#) et que le plugin standard "volet des tags" est également activé, il est possible de renommer des tags : toutes les notes où ce tag apparaît arboreront le tag avec son nouveau nom.

Ce plugin permet aussi d'affiner la recherche en incluant ou excluant certains tags.

3.5 autres langages (html, LaTeX)

Le markdown est une syntaxe très simple à apprendre et très versatile. Toutefois, il n'a pas été conçu comme une syntaxe universelle permettant d'éditer intégralement un ouvrage ou un diaporama. Comme l'indiquait John Gruber (voir plus haut [Mode d'emploi d'Obsidian > différentes saveurs de Markdown](#) pour obtenir la présentation souhaitée pour certains contenus, on aura besoin parfois de faire appel à d'autres langages dans son document, notamment le HTML ou LaTeX (si vous avez téléchargé un éditeur LaTeX vous pourrez constater que le LaTeX et le Markdown font plutôt bon ménage dans Obsidian)

utilisation du LaTeX

Obsidian n'est pas un éditeur en LaTeX et ne se substitue pas à ce genre d'éditeur.

Toutefois, il est possible de rédiger des formules mathématiques et chimiques directement dans Obsidian, comme on le ferait dans un document en LaTeX, grâce à la bibliothèque Mathjax qui a été intégrée aux dernières versions d'Obsidian.

formules mathématiques

Le package mathjax a été chargé dans le code d'Obsidian [en septembre 2021](#). Ce package permet d'utiliser les fonctionnalités de la bibliothèque mathjax sur un document en markdown. Parmi ces fonctionnalités, on trouve également la [syntaxe de base du mhchem](#) pour les formules chimiques.

Une version étendue de mathjax est disponible [sous la forme d'un plugin](#)

Grâce à cela on peut encapsuler des équations lisibles en LaTeX en les encadrant par le signe \$

π s'écrit ainsi `π`

`$67\sqrt{\prod{67}^{\{4\}}\rightarrow \mathrm{C}\{56\}^{\{'\}}$`

donne :

$$67\sqrt{\prod_{67}^4} \Rightarrow C'_{56}$$

Cette formule ne correspond à rien, elle a été créée à partir d'un [éditeur de formules en LaTeX](#) disponible en ligne.

Pour centrer l'équation (au lieu de l'insérer dans le texte, redoubler le \$)

$$67\sqrt{\prod_{67}^4} \Rightarrow C'_{56}$$

donne :

$$67\sqrt{\prod_{67}^4} \Rightarrow C'_{56}$$

Par rapport à la taille de défaut, les caractères présents dans ces formules issues de MathJax ont une taille augmentée. Cela a été réglé en ajoutant une instruction concernant tous les caractères édités avec MathJax dans le CSS de base (pour les questions relatives au CSS, [voir plus bas](#))

Voici ce qui a été ajouté en l'occurrence :

```
.MJX-TEX {  
  font-size: 180%;  
}
```

[source](#)

Utiliser LaTeX pour réaliser les exposants :

- le mebioctet ou Mi parfois écrit Mio correspond à 2^{20} octets
- le gibioctets ou Gi parfois écrit Gio correspond à 2^{30} octets

donne :

- le mebioctet ou Mi parfois écrit Mio correspond à 2^{20} octets
- le gibioctets ou Gi parfois écrit Gio correspond à 2^{30} octets

formules chimiques

La formule de l'eau peut-être notée de la manière suivante : H_2O cela donne H_2O

Etat d'oxidation : $\text{Fe}^{\text{II}}\text{Fe}^{\text{III}}_2\text{O}_4$

([source](#))

Mise en page

Le markdown ne donne pas vraiment de solution lorsqu'on veut mettre deux images en regard (on ne peut pas encapsuler ces images dans un tableau). On est obligé pour réaliser cela de recourir au package graphicx de LaTeX.

```
\begin{figure}
\begin{minipage}[t]{.4\linewidth}
  \begin{center}
    \includegraphics[width=6cm]{drake1.png}
    \caption{pyramide ideale}
  \end{center}
\end{minipage}
\hfill
\begin{minipage}[t]{.4\linewidth}
  \begin{center}
    \includegraphics[width=6cm]{drake2.png}
    \caption{pyramide effective}
  \end{center}
\end{minipage}
\end{figure}
```

utilisation du HTML

Lorsqu'on copie-colle du html dans Obsidian, généralement ce html est copié avec des balises supplémentaires (``).

Pour copier-coller du HTML qui fonctionne correctement, préférer Ctrl+shift+V à Ctrl+V

faire un épigraphe

Cela fonctionne en vue d'une conversion avec [pandoc](#)

créer un épigraphe

```
<div class='epigraph'>

*"All human things are subject to decay, and when fate summons, Monarchs
must obey."* --- John Dryden.

</div>
```

pour faire des tableaux avec des colonnes ou des lignes fusionnées

code

```
<table>
  <thead>
```



```

<tr>
  <th>entête 1</th>
  <th>entête 2</th>
  <th>entête 3</th>
</tr>
</thead>
<tbody>
  <tr>
    <td rowspan=4>valeur 1 (4 cellules verticales)</td>
    <td rowspan=2>valeur 2 (2 cellules verticales)</td>
    <td>valeur 4</td>
  </tr>
  <tr>
    <td>valeur 5</td>
  </tr>
  <tr>
    <td rowspan=2>valeur 3 (2 cellules verticales)</td>
    <td>valeur 6</td>
  </tr>
  <tr>
    <td>valeur 7</td>
  </tr>
</tbody>
</table>

```

Résultat

entête 1	entête 2	entête 3
valeur 1 (4 cellules verticales)	valeur 2 (2 cellules verticales)	valeur 4
		valeur 5
	valeur 3 (2 cellules verticales)	valeur 6
		valeur 7

Si on veut fusionner des cellules sur plusieurs colonnes, plutôt que sur plusieurs lignes, utiliser l'attribut *colspan* comme dans [cet exemple](#)

Mois	Economie
Janvier	100€
Février	80€
Somme: 180€	

Pour changer la couleur du texte, on peut recourir à du html, par exemple :

```
<font color="#cc241d">Texte en couleur</font>
```

donne :

Texte en couleur

faire un saut de page nécessite l'inclusion de html

```
<div style="page-break-after: always visibility: hidden">  
\pagebreak  
</div>
```

Le contenu qui suit se trouvera sur la page suivante.

\pagebreak

pour passer une ligne (en vue d'une conversion en PDF avec Pandoc)

\newline

dimensionner des images

L'image peut prendre trop d'espace sur la page. Dans ce cas, il convient de déterminer sa dimension pour qu'elle s'insère sans problème sur la page. Cela se fait au moyen des attributs HTML width ou height exprimés en valeur absolue (nombre de pixels ou relatives (pourcentage par rapport à la largeur ou à la hauteur de l'image initiale)).

```
![bibliotheque](images/buvs.jpg){ width=20% }
```

réduit à 20% la largeur initiale de l'image (on ne touche pas à la hauteur pour garder les proportions).



{ width=20% }

```
{ width=400 fig-align="center"}
```

on peut évidemment définir cette largeur en pixels :

```
![bibliotheque](images/buvs.jpg){ width=250px }
```

réduit la largeur de l'image à 100 pixels



{ width=250px }

Quant à l'alignement du texte, cela peut se traiter avec des lignes de HTML insérées dans le document ou bien en renvoyant vers un fichier CSS, [comme le suggère cet internaute](#).

commenter un document



Supposons que l'on souhaite laisser des commentaires pour orienter le sens de l'édition d'une note, sans que ce commentaire apparaisse dans la version finale. Bien qu'Obsidian ne soit pas d'abord un éditeur de texte, on peut utiliser la fonction *comment block code* présente dans les principaux éditeurs de texte pour commenter du code mais aussi du texte en markdown

voici le texte à commenter <!-- commentaire de ce texte -->

inclure des graphiques dans les notes

la bibliothèque en javascript mermaid.js permet d'inclure des graphiques (camemberts, diagrammes, flow charts, etc) dans une note.

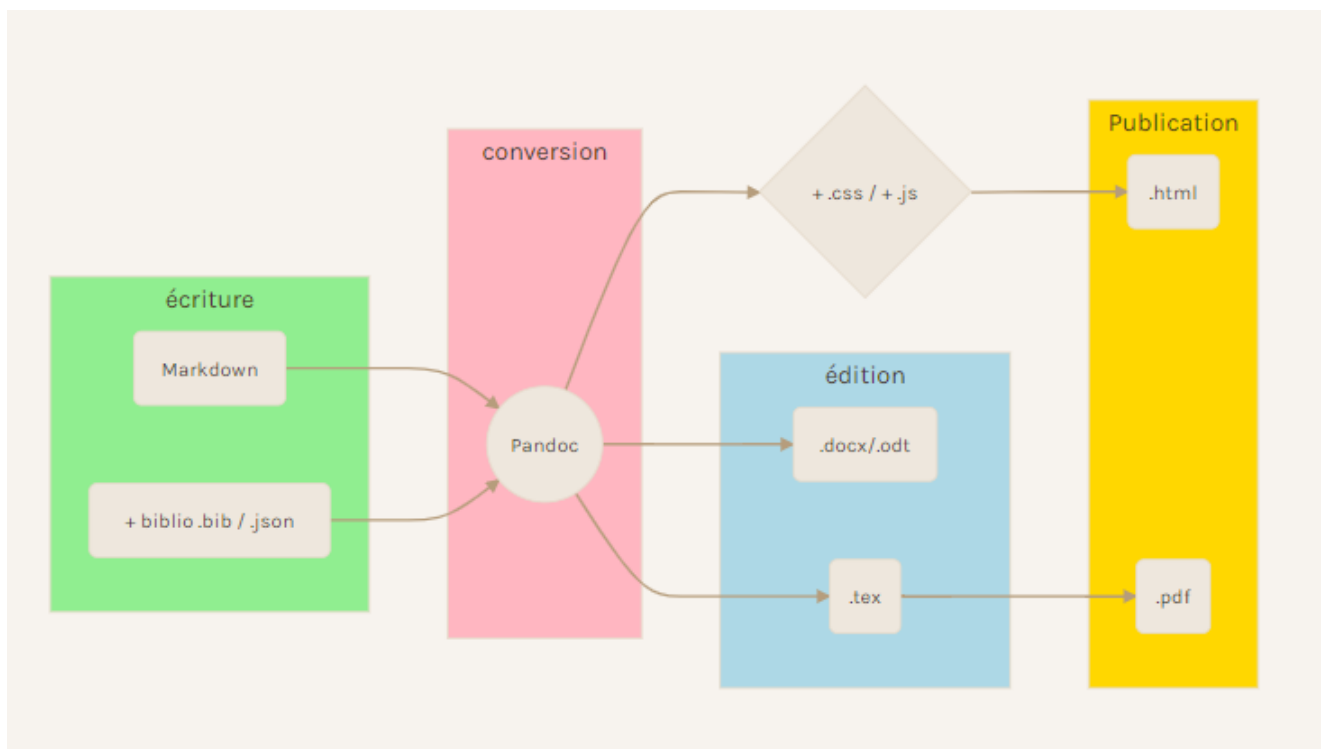
Il existe de nombreux tutos consacrés à cet outil qu'il n'est pas nécessaire d'installer dans Obsidian. Voir ce [guide](#) entre autres. Pour s'essayer à la construction de ces graphiques, on peut utiliser [le site https://mermaid.live](https://mermaid.live)

exemple : [worklfow éditorial réalisé par UjuBib](#) :

```

38
39 ```mermaid
40 graph LR
41
42     subgraph écriture
43         id1(Markdown)
44         id7(+ biblio .bib / .json)
45     end
46
47     subgraph conversion
48         id2((Pandoc))
49     end
50
51     subgraph édition
52         id3(.docx/.odt)
53         id4(.tex)
54     end
55
56     subgraph Publication
57         id5(.html)
58         id6(.pdf)
59     end
60
61     id8{+ .css / + .js}
62
63     id1→id2
64     id7→id2
65     id2→id3 & id4
66     id2→ id8 →id5
67     id4→id6
68
69     style écriture fill:lightgreen
70     style conversion fill:lightpink
71     style édition fill:lightblue
72     style Publication fill:gold
73
74 ```

```

3.6 importer des notes écrites dans un autre format que Markdown

Supposons que j'aie déjà accumulé avant de connaître Obsidian des notes en texte simple (format TXT) ou en format Word (traitement de texte LibreOffice ou Word), comment faire pour les convertir en masse depuis le répertoire où elles se trouvent. Les notes à importer dans Obsidian doivent être en markdown. Comment réaliser cette conversion de mes notes en TXT vers ce format markdown ?

Il existe plusieurs manières de convertir un ensemble de documents d'un format à l'autre. On peut utiliser un terminal de commandes (bash) pour ce faire. Par exemple sous Linux, si on ouvre un terminal dans le répertoire où se trouvent les notes à convertir, on entrera la commande suivante :

Sur Windows, on peut utiliser le terminal de commande Powershell pour réaliser cette conversion directement en bash, ou bien en combinant cette commande avec le convertisseur [Pandoc](<https://pandoc.org/installing.html>). La seconde option est détaillée ci-dessous:

```
for /r %i in (*.txt) do multimarkdown -b %i
```

Pour MacOS, on peut se contenter de [changer les extensions des fichiers](#)

Il est toujours pratique d'avoir installé sur sa machine lorsqu'on utilise des documents en markdown qui peuvent être convertis dans une multitude de formats différents. Nous supposons que Pandoc est installé. Ouvrir un terminal Powershell dans le répertoire où se trouvent les notes et entrer la commande suivante :

```
Get-ChildItem . -Filter *.txt |  
Foreach-Object {  
    pandoc $_ -o $_.Name.Replace('.txt', '.md')  
}
```

Puis exporter toutes les notes en markdown dans le répertoire des notes sous Obsidian.
Les noms de fichiers deviendront les titres des notes (voir [cette vidéo](#) sur Youtube).

Reste à faire les liens entre les différentes notes ce qui est un travail bien plus long que la simple conversion que nous venons de faire.

4. Les plugins d'Obsidian

La richesse d'Obsidian tient au nombre en croissance rapide des plugins qui rendent cet outil adaptable à un grand nombre de tâches.

Il existe deux types de plugins :

- Les plugins par défaut
- Les plugins communautaires

Les premiers sont installés en même temps qu'Obsidian, il ne reste plus qu'à les activer si on souhaite s'en servir.

Les seconds nécessitent la désactivation du mode sans échec (aller dans les paramètres > plugins tiers > désactiver le mode sans échec).

A partir de là, pour trouver un plugin communautaire :

- soit on passe par le catalogue des plugins depuis Obsidian (parcourir la liste des plugins communautaires)
- soit on va chercher le plugin sur github, là où il a été déposé par le concepteur. C'est sur github, par exemple, qu'on doit aller chercher le plugin qui permet de faire un export d'une note depuis Obsidian en PDF à l'aide de Pandoc.

Quelque soit le type de plugin, on peut paramétrer leur utilisation dans les paramètres une fois qu'il a été installé et activé (dans les Paramètres, descendre jusqu'au menu *option des plugins*)

Il est très utile de paramétrer des [raccourcis-clavier](#) pour chaque fonctionnalité appartenant ou non à un plugin. Cela fait gagner beaucoup de temps au quotidien (paramètres > options > raccourcis-clavier ; chercher la fonctionnalité par son nom et lui associer une combinaison de touches ; par exemple créer un graphique local à partir de la note ouverte = shift+Ctrl+G).

chaque plugin chargé est associé à un coffre en particulier et si vous avez plusieurs coffres, il faudra charger à l'intérieur de chacun d'eux un même plugin. Cela se voit bien au niveau de l'arborescence Windows : le logiciel Obsidian est chargé par défaut au niveau du C:/Programmes, les coffres sont hébergés ailleurs (où l'utilisateur le souhaite), dans ces coffres, les plugins se trouvent sous .obsidian > plugins

4.1 Les plugins par défaut

Les plugins par défaut font partie du coeur d'Obsidian. Pour les activer les utiliser, aller dans les paramètres (icône roue) et sélectionner les *plugins principaux*

Seuls deux plugins correspondent à une option payante d'Obsidian : la publication des notes sur le serveur d'Obsidian et leur synchronisation avec les notes hébergées en local.

Voici une revue de quelques plugins installés par défaut :

explorateur de notes

Par défaut, trois plugins sont activés sur Obsidian pour faciliter la recherche :

- l'explorateur de fichier (volet latéral gauche qui permet de naviguer dans une arborescence de fichiers)
- le plugin rechercher permet dans le même volet de rechercher par catégorie : file(), tag(), path()...
- le selecteur rapide : la commande Ctrl+O (commande par défaut) permet d'accéder directement à tous les fichiers en tapant un élément de leur nom dans un champ.

Explorateur de fichiers

L'explorateur de fichiers permet de naviguer dans les fichiers de votre coffre à l'endroit où celui-ci est hébergé.

Vue graphique globale

La vue graphique vous permet de visualiser l'ensemble de vos notes et des liens qui les unissent sous la forme d'un graphe.

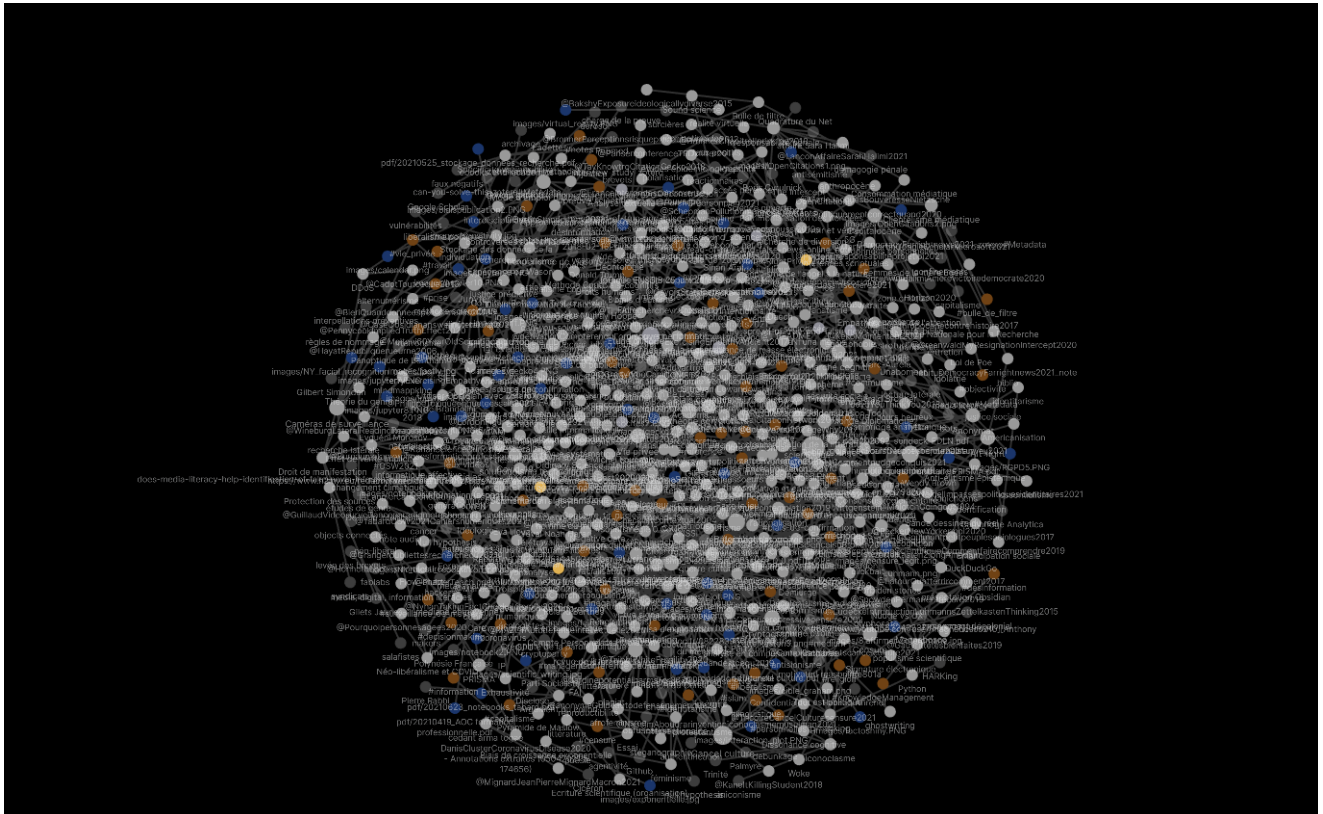
D'après Arthur Perret [@PerretWhatpointgraph2022](#), le bénéfice de ces visualisations qui passent aux yeux de certains comme des gadgets dont on peut se passer n'est pourtant pas à négliger :

- le graphe nous rappelle que notre mode d'écriture et de mémorisation est par défaut réticulaire. Ce que nous écrivons de nouveau s'ancre toujours dans un savoir déjà acquis et des notes déjà prises. Faire des liens avec ce savoir pré-existant permet de solidifier ses connaissances

- le graphe nous permet de prendre conscience de sujets émergents dans notre prise de notes et de liens qui pourraient être faits entre ces sujets et d'autres plus anciens.
- le graphe nous permet de trouver de l'information reliée à la note que nous sommes en train d'écrire et que nous ne cherchions pas (**sérendipité**). Créer des liens entre des notes voisines permet de sortir la note de son isolement, cela empêche la pensée en silo :

A graph is a memory aid that prevents siloed thinking.

Lorsque les notes ne sont pas liés à quelques projets, cette constellation devient vite illisible.



Bien entendu, on peut zoomer sur telle ou telle partie, mais ça devient vite confus au bout d'un moment.

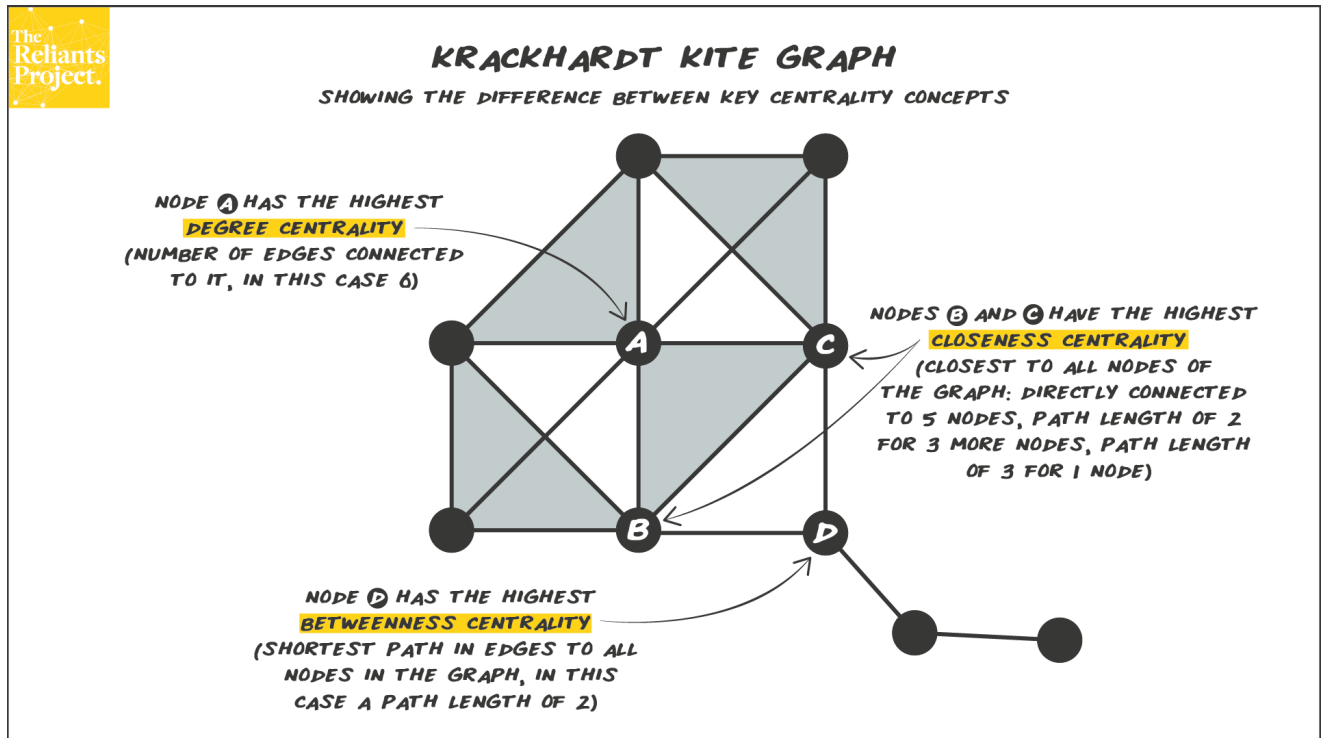
pour visualiser les notes connexes à la note de son choix, il suffit de laisser la souris sur cette note

d'autres (et peuvent donc passer pour plus importantes que d'autres pour l'esprit qui a collationné ces notes). Il existe trois types de centralité :

la centralité de degré : la note qui a le plus de liens avec des notes qui ont elles-mêmes beaucoup de liens

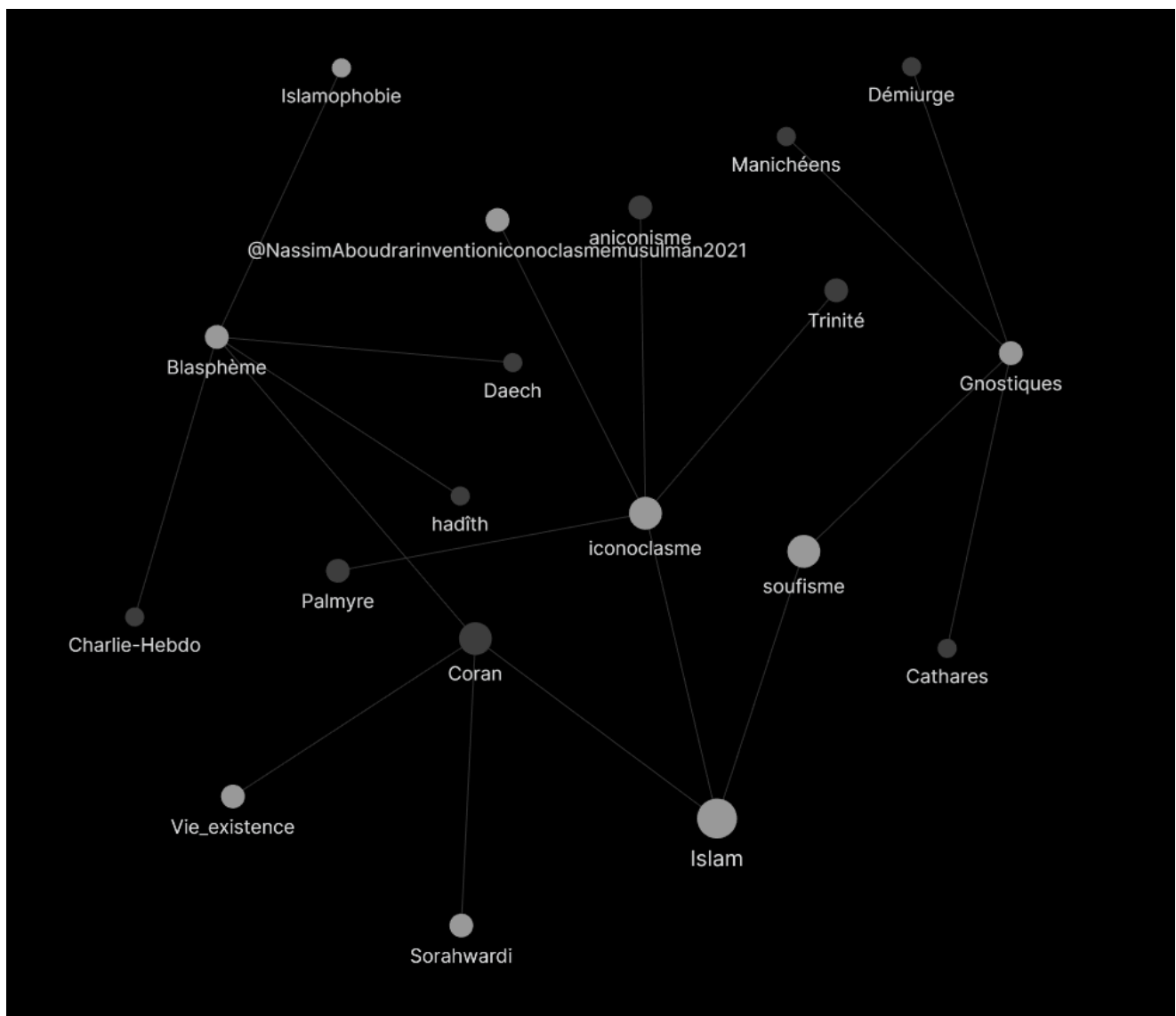
la centralité de proximité : la note à partir de laquelle les chemins sont les plus courts vers l'ensemble des notes du graphe

la centralité d'intermédiation : la note qui relie entre elles le plus de pôles (typiquement une note orange pourrait relier un pôle de notes fruits et un pôle de notes objets de couleur orange)



vue graphique locale

Plus intéressante est la possibilité d'ouvrir un graphe local à partir de la note (aller dans les raccourcis clavier pour paramétrer la combinaison de touches qui vous conviendra pour faire apparaître ce graphe)



On peut régler la taille du graphe en choisissant le nombre de degrés à partir du noeud que constitue la note affichée (de 1 à 6 degrés - ci-dessus, le nombre de degrés est fixé à 3 : "demiurge" est distant de 3 degrés de "islam")

Comme dans les outils de visualisation sous forme de graphe (par exemple Gephi), on peut régler la force du graphe, la taille des noeuds, caractériser les liaisons au moyen de flèches (quelle note cite quelle autre note)...

Les rétroliens

les rétroliens ou *backlinks* constituent la grande force des applications comme Roam, org-roam ou Obsidian. Quelque soit la note que vous affichez, si vous activez ce plugin, vous pourrez vous rendre facilement aux notes qui font référence à celle-ci au moyen des rétroliens en bas de page.

Modèles

Activer ce plugin, vous permettra de constituer des templates (voir ci-dessous)

Diapositives

Il est très simple grâce à ce plugin de constituer des diapositives et les présenter à la place d'un document impress ou powerpoint.

chaque diapositive est séparée de la précédente par trois tirets (---)

Les plugins payants Sync et Publish

Ces plugins sont payants parce qu'ils supposent un hébergement des données sur les serveurs d'Obsidian et une synchronisation avec entre ces serveurs et l'ordinateur qu'on utilise pour modifier les fichiers.

Exemple de publication d'un coffre Obsidian avec Publish : le carnet de notes de [Yann Houry](#)

Leur recours peut-être intéressant pour constituer des tutoriels accessibles en ligne à partir de ses notes (voir par exemple ces [50 leçons et exercices pour apprendre le HTML, le Javascript et le CSS](#))

5. Créer des modèles (templates)

Les templates se trouvent dans un répertoire particulier à la racine du coffre.

Rédiger un template en markdown et l'enregistrer à cet endroit

Paramétrer un raccourci clavier pour faire apparaître le menu de sélection des templates.

5.1 Utilité des templates et des champs de métadonnées

Un template peut avoir des champs pré-remplis à l'instar d'une fiche :

lieu du rendez-vous ou lien de la visio :

date du rendez-vous :

heure du rendez-vous :

participants :

sujets :

ou encore (par exemple pour une note de lecture)

titre de l'ouvrage :

auteur :

année d'édition :

résumé :

mon appréciation personnelle sur l'ouvrage :

intérêt par rapport à mon travail :

Il peut s'agir aussi d'une configuration habituelle qu'on veut donner à nos notes, par exemple, cette configuration peut prendre en charge le pavé YAML

5.2 YAML

...n'est pas seulement un nouveau format de balises comme le veut son acronyme (yet another markup language). Il s'agit d'un espace qui va recueillir toutes les métadonnées importantes du document.

Le YAML frontmatter (frontispice ou pavé YAML) se présente en tout début de document (ne rien entrer avant) et est délimité par deux séries de trois hyphens (tirets moyens)

exemple de pavé YAML

```
---
title: Mode d'emploi d'obsidian
subtitle: Comment prendre efficacement des notes pour sa thèse
date: 21/06/2021
author: Damien Belvèze
affiliation: Université de Rennes 1
toc: true
toc-title: table des matières
link-citations: true
bibliography: mabiblio.bib
biblio-style: nature.csl
tags:[tag1, tag2, tag3]
aliases:[alias1, alias2, alias3]
---
```

ajouter une affiliation en utilisant des packages de LaTeX :

```
---
title: Mode d'emploi d'obsidian
subtitle: Comment prendre efficacement des notes pour sa thèse
date: 21/06/2021
author: Damien Belvèze
affiliation: Université de Rennes 1
toc: true
toc-title: table des matières
link-citations: true
bibliography: mabiblio.bib
biblio-style: nature.csl
tags:[tag1, tag2, tag3]
aliases:[alias1, alias2, alias3]
header-includes: |
\usepackage{authblk}
\author{John Doe \thanks(john.doe@email.com) }
```

Source

- les valeurs inscrites en tant que titre, sous-titre, nom de l'auteur et date, vont au moment de la conversion avec Pandoc constituer la page de titre
- si toc (table of content) reçoit la valeur *true*, une table des matières sera générée.
- si *link-citations* reçoit la valeur *true*, les liens seront cliquables dans le format de sortie (html ou pdf)
- les champs bibliography et biblio-style règlent les aspects bibliographiques.
- *bibliography*: spécifier le chemin absolu (C:/.../fichier.bib) ou relatif (references/fichier.bib) vers le document qui contient les références bibliographiques (un fichier .bib exporté de [Zotero](#) par exemple)
- *biblio-style*: lien vers le fichier csl qui contient les règles de présentation de la bibliographie (par exemple [nature.csl](#))
- *tags*: permet d'ajouter des tags à la note. Ceux-ci ne seront pas visibles dans le document issu de la conversion
- *aliases*: cette fonctionnalité ajoutée à la version 0.9 est très importante : elle permet de lier les notes entre elles sans pour autant respecter exactement la forme sous laquelle se présente le titre de la note.
Par exemple, dans une note intitulée "féminisme", ajouter comme alias *féministe*, *féministes*, *féminismes* permettra de faire un lien vers cette note à partir d'autres notes où apparaîtront les noms *féminisme* (au singulier et au pluriel) et *féministe* (au singulier et au pluriel). Cela permet également de gérer la polysémie de certains termes dans l'organisation de ses notes.

5.3 paramétrer des modèles par défaut

on a vu plus haut comment créer des modèles. Si on en reste à cette étape, à chaque fois qu'on créera une note, il faudra aller choisir les modèles dans la galerie de modèles existants que l'on aura créés.

Comment faire pour que la note créée prenne immédiatement un modèle par défaut ?

Le plugin **Templater** rend cela assez facile :

- charger le plugin, l'activer
- dans les paramètres du plugin, indiquer où se trouve le dossier qui contient les modèles
- mettre l'option "trigger templater on new file creation" en position *On*
- dans le paramètre *empty file template* pointer vers le modèle de son choix : c'est ce modèle qui s'appliquera désormais à chaque création de note.

Ce n'est que la base du plugin **templater**. Ce plugin permet d'éditer en javascript des préférences pour appliquer automatiquement des modèles différents en fonction de ce que contient le nom de la note.

templater permet également d'inscrire des variables et des fonctions en javascript dans une note.

A titre d'exemple, la fonction

```
<% tp.file.creation_date("DD-MM-YYYY") %>
```

inscrit la date du jour dans la note. C'est une commande très utile à insérer dans un template afin de permettre à toutes les notes créées à partir de ce template de contenir la date du jour.

La fonction

```
<% tp.file.title %>
```

Plus de fonctions [ici](#)

reprend le nom du fichier à l'endroit souhaité (par exemple devant l'attribut title dans le bloc YAML) comme dans l'exemple ci-dessous :

```
title: <% tp.file.title %>
subtitle:
author: Damien Belvèze
date: <% tp.file.creation_date("DD-MM-YYYY") %>
link_citations: true
bibliography: mylibrary.bib
biblio_style: csl\ieee.csl
aliases: []
tags: []
```

6. Inscrire son quotidien dans Obsidian

Si l'on souhaite que sa bibliothèque de notes dans Obsidian ne reflète pas seulement les documents et concepts relatifs à sa thèse, mais qu'on veut y intégrer aussi des éléments de son organisation personnelle (todo list, notes journalières), Markdown permet de faire cela avec quelques plugins complémentaires.

6.1 Todolists

Obsidian n'est pas principalement conçu pour organiser ses tâches et gérer des *todo lists*, mais on peut néanmoins adapter cet outil à cette fin dans une certaine mesure.

- tâche à effectuer avant le 6 décembre 2021

- [] tâche > 2021-12-6

☐ tâche > 2021-12-6

- tâche effectuée

- [x] tâche > 2021-12-6

☒ tâche > 2021-12-6

chercher dans l'explorateur une tâche à faire :

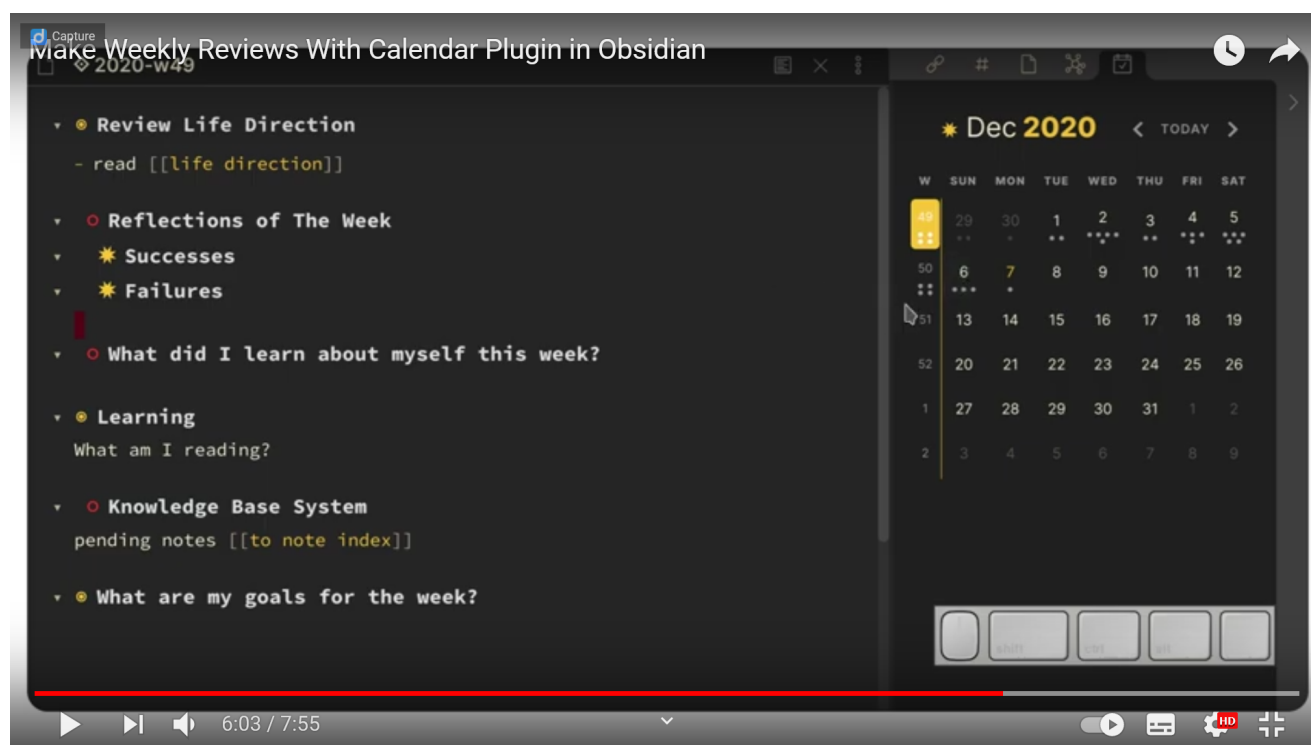
"- [*]"

6.2 calendars

On peut afficher un calendrier dans Obsidian au moyen du plugin communautaire Calendar. Si cela fait sens pour vous, activer dans les paramètres du plugin calendar le nombre des semaines car il est désactivé par défaut.

Cliquer sur une journée permet de créer une note relative à cette journée.

Il est possible de créer un template pour les notes quotidiennes en indiquant son chemin dans les paramètres du plugin.



6.3 daily notes

Obsidian vous permet si vous le souhaitez de créer des notes quotidiennes.

Dans les plugins par défaut, mettre le plugin "notes quotidiennes" sur *on*

7. Utiliser Zotero avec Obsidian

module Citations de Jon Gautier

Prérequis : disposer de Zotero avec le plugin BetterBibTeX chargé.

Avoir téléchargé Pandoc et un éditeur LaTeX sur sa machine.

Si la version de Pandoc dont on dispose n'inclue pas le plugin pandoc-citeproc, charger ce plugin.

1. Charger dans Obsidian le [plugin Citations](#) qui permet d'importer des données dans Zotero
2. Comment charger un plugin dans Obsidian ?
 - Paramètres > Plugins Tiers > désactiver le mode sans échec
 - Pour ce plugin, chercher avec "Citations"
 - installer ce plugin
3. Créer dans Obsidian votre espace de travail
 - créer un dossier (intitulé Article)
 - créer une note (qui deviendra le futur article à publier)
4. Dans Zotero exporter la collection qui correspond aux références de l'article en la tenant à jour (appelons ce fichier references.bib). Enregistrer le fichier dans le dossier article sous un répertoire qu'on crée à l'occasion (par exemple en l'intitulant 'references')
5. Dans les options du plugin, indiquer le nom du répertoire ('références') qui contient le fichier .bib provenant de Zotero. Indiquer le chemin relatif vers le fichier bib :
/espacedetravail/references/references.bib
Si le chemin est le bon, le nombre d'enregistrements du fichier bib devrait apparaître. Il faut également définir un dossier où iront les références bibliographiques issues de Zotero (notes précédées d'un @). Si on souhaite que celles-ci aillent à la racine du coffre, on supprimer le texte qui est par défaut. Si on souhaite les ranger dans un dossier particulier, créer ce dossier (par exemple sous .obsidian) et indiquer au plugin le chemin vers ce dossier.
6. Dans les Paramètres d'Obsidian > raccourcis clavier, paramétrer un raccourci clavier pour insérer une citation en format markdown (Citations: insert markdown citation). Saisissons à cet endroit par exemple la combinaison de chiffres suivante : **Shift + Ctrl + M**
7. Dans l'article, pour insérer une citation, utiliser le raccourci clavier indiqué plus haut.
8. Une fois toutes les références ajoutées, ouvrir dans l'explorateur le répertoire où se trouve le fichier dans lequel on a inséré ces références.
9. Ouvrir un shell, taper la commande permettant à Pandoc de convertir ce document markdown en document PDF en utilisant le filtre pandoc-citeproc
10. On obtient le PDF souhaité avec les références correctement représentées (appels de citation et bibliographie) selon le style choisi dans la commande pandoc.

module Zotero integration

(source: [@dannytalkstechZoteroObsidianIntegration2022](#))

8. importer des notes créées ailleurs dans Obsidian

MDnotes (plugin de Zotero)

il est possible d'importer des notes créées dans Zotero avec le plugin MDnotes

1. Charger le [plugin MDnotes](#) dans Zotero
2. paramétrer le plugin

Mdnotes preferences

Export preferences | File names

☒ Use the item's citekey as title?

☐ Show Standalone Note menu

File organization:
Exporting notes as a single file or in multiple files

☒ Single file ☐ Split files

Internal Links
The link format is applied to all the metadata fields, except the cite key.

☒ No links ☐ [[Wiki style]] ☐ [Markdown style](markdown-style)

No links will output metadata as plain text
Markdown links will change how files are renamed and substitute spaces with dashes.

Export directory

Which path to suggest in the file picker

☐ Attach file links to Zotero

Template folder

Path where templates are stored. Templates should be stored in the root folder; subfolders are not supported.

3. Indiquer un répertoire d'export dans Obsidian (par exemple un répertoire intitulé MDnotes)
4. Dans ce répertoire, MDnotes cherchera les fichiers suivants :

- Mdnotes Default Template.md
- Standalone Note Template.md
- Zotero Metadata Template.md
- Zotero Note Template.md

pour paramétrer l'affichage de la note de la façon souhaitée, pour une note fille, éditer le document MDnotes Default Template.md, si c'est une note parente Standalone Note Template.md

Pour exporter une note, clic droit sur la référence > MDnotes > export to markdown

Le fichier sera exporté en markdown à l'endroit spécifié dans le menu du plugin

hypothes.is

Si on utilise hypothes.is pour annoter le web (plugin actuellement accessible sur Chrome), il existe un plugin d'Obsidian qui permet de synchroniser ses notes sur hypothes.is avec un répertoire d'Obsidian.

1. Charger le plugin hypothesis (disponible dans les plugins communautaires)
2. activer-le (ne pas oublier cette étape !)
3. dans hypothesis, aller sur developper, générer un token
4. dans le plugin d'Obsidian, spécifier le répertoire où doivent aller les notes et entrer le token

Dès lors que le plugin est chargé, un bouton dans le menu latéral permet de synchroniser les notes prises dans Hypothes.is avec Obsidian

exécuter du code dans une note

On peut aussi importer du code dans une note et l'exécuter au moyen d'un plugin spécifique "[execute code](#)". Ce plugin fait partie de la liste des plugins communautaires.

Une fois chargé et activé, il permet d'exécuter le code fourni dans les *code blocks* :

```
name = input("What's your name? ")
print("Hello " + name + ", have a good day!")
```

9. changer le CSS

une feuille de style CSS va définir pour un document en markdown comment va se présenter un titre 1 (# titre1), un titre 2 (## titre2), comment une image va être alignée par rapport au texte, comment les citations vont être présentées.

Ce CSS peut être pris en compte lorsqu'on exporte la note d'obsidian vers un fichier HTML ou bien lorsqu'on visualise une note (aperçu en mode preview ou publication sur les

serveurs d'Obsidian avec le plugin payant Publish)

Si aucun chemin vers un fichier CSS n'est spécifié dans les métadonnées (YAML), Obsidian va appliquer la feuille de style CSS qui se trouve à la base du répertoire :

- dans les paramètres d'Obsidian (apparence), vérifier que l'option *appliquer un css personnalisé* est cochée
- ouvrir un éditeur de texte (bloc-note, notepad++, etc.) et éditez le CSS dont vous avez besoin.
- Sous le répertoire .obsidian, créer un dossier intitulé *snippets*. Enregistrer le fichier css dans ce dossier. Au besoin, recharger le css depuis le menu apparence et vérifier que l'utilisation du CSS ajouté est bien activée. Attention : les changements ne seront pas visibles en édition. Le CSS n'affecte que le mode lecture.

A titre d'exemple, le code...

```
h1 { color: #fe801a; } h2 { color: #d79920; } h3 { color: #689d6a; }  
...règle la couleur des titres.
```

Il n'est pas question ici de créer un guide d'apprentissage du CSS, mais on peut trouver [ici](#) des thèmes tout faits pour les notes et [là](#) des *snippets*, autrement dit des bouts de code en CSS qui permettent d'obtenir un affichage personnalisé de certains éléments. Ces exemples peuvent être repris pour constituer une feuille de style CSS qui nous soit propre.

10. Conversion de la note en PDF ou dans un autre format de sortie

On peut éditer une note en PDF en utilisant la fonctionnalité interne d'Obsidian (exporter en PDF).

En revanche, pour gérer certains aspects comme la bibliographie, on aura besoin d'utiliser Pandoc en lien avec LaTeX

10.1 pandoc

Pandoc est le couteau-suisse de l'édition numérique. Il ne dispose pas à ce jour d'une interface graphique, mais les manuels d'utilisation ne manquent pas pour apprendre à taper les lignes de commande qui vous permettront d'obtenir le résultat souhaité.

avec le plugin obsidian-pandoc

L'export du document de son format en markdown vers un format html, wiki, odt, doc, etc. Peut se faire avec pandoc et le plugin obsidian pour pandoc seuls.

En revanche, pour faire une conversion et un export d'un document en md vers un PDF, il faut en outre télécharger LaTeX, ce qui peut prendre du temps.

On part du principe que LaTeX est installé sur la machine :

1. installer la dernière version de pandoc

Il est recommandé d'aller prendre le fichier zip de la dernière version disponible sur [github](#)
dézipper le fichier sur le bureau

Afficher dans l'explorateur les fichiers cachés pour faire apparaître *appdata* dans le répertoire de l'utilisateur

aller à C:user/username/.appdata/local/

couper le dossier pandoc sur le bureau et copier le dans .appdata/local/

vérifier que pandoc.exe fait partie des fichiers collés ainsi que pandoc-citeproc

2. Installer dans Obsidian le plugin

Prendre le code source sur [github](#)

dézipper l'archive le sur le bureau. Aller à l'intérieur du répertoire, copier le dossier et le coller dans le répertoire du coffre (obsidian > plugins), couper et coller ici le dossier obsidian-pandoc.

Le répertoire à coller à cet endroit contient directement les fichiers du plugin dont le fichier main.js

Comme pour les autres plugins, ne pas oublier de l'activer dans Obsidian dans la liste des plugins communautaires.

3. Paramétrer le plugin

Définir avec un chemin absolu (D:/...Vault/export_pdf) un dossier dans lequel vont aller tous les fichiers exportés et convertis dans le format voulu

Attention : même si LaTeX est correctement chargé sur la machine, l'export en PDF ne fonctionnera pas avec une version de Pandoc antérieure à 2.7. Sous Linux avec la commande apt-get, ce n'est pas toujours la dernière version ni même une version récente qu'on charge. Avec la commande pandoc -V bien vérifier qu'on a chargé la dernière version de Pandoc.

Définir des raccourcis clavier pour les différents formats d'export

Par exemple Ctrl+Alt+P pour un export en PDF, Ctrl+Alt+W pour un export en .doc, etc.

10.2 Directement avec Pandoc

Ouvrir dans le dossier où se trouve la note le menu contextuel qui permet d'accéder à Powershell (sous Windows : shift + clic-droit) ou ouvrir le terminal à cet endroit sous Linux.
Pour Windows.

script avec Powershell

- Windows : ouvrir l'interface de commande powershell

Il peut être nécessaire avant de convertir le document md en pdf de supprimer les liens internes en markdown (liens vers d'autres notes), sinon le PDF comportera des [[]] indésirables dans le texte final.

On peut utiliser powershell pour cela :

```
((Get-content -path "C:\\lien\\vers\\le\\fichier\\source.md" -Raw).replace("\n",')).replace("]", "'') | Set-Content -Path "C:\\lien\\vers\\le\\fichier\\destination.md"
```

Ajouter à cela la commande propre à pandoc.

Cela donne par exemple pour un fichier intitulé note.md

```
(Get-content ./note.md -Raw).replace("[", "").replace("]", "") | Set-content ./note2.md | pandoc ./note2.md --bibliography C:\Users\dbelveze\Nextcloud\obsidiantest\biblio\mylibrary.bib --csl C:\Users\dbelveze\Nextcloud\obsidiantest\csl\nature.csl --pdf-engine=xelatex --citeproc -f markdown+smart -o note.pdf | Remove-Item ./note2.md
```

ce programme:

- affiche le contenu de note.md en supprimant les [et les]
- crée à partir de ce texte un nouveau fichier
- convertit ce nouveau fichier markdown en pdf en parsant les références présentes dans un fichier .bib défini, selon un style csl défini

On peut faire de ce programme un script powershell (script.ps1) et l'exécuter dans powershell ./script.ps1

```
pandoc 'document.md' --from html --to pdf --pdf-engine=xelatex --citeproc -f markdown+smart -o document.pdf
```

--pdf-engine=xelatex : utilise le programme LaTeX que vous avez installé sur votre machine pour convertir le document en PDF

--citeproc -f markdown+smart : gère les références bibliographiques

(prévoir un titre 1 bibliographie à la fin du document)

script avec Python

possibilité de convertir des fichiers avec le package [pypandoc](#)

```
import os
import pypandoc
fichier = input("Quel fichier voulez-vous convertir ?")
with open(fichier, 'r') as file :
```

```

filedata = file.read()

# Replace the target string
filedata = filedata.replace('[[', ' ').replace(']]', '')

# Write the file out again
with open('file.txt', 'w') as file:
    file.write(filedata)
# convert modified file into open document file
pypandoc.convert_file('file.txt', format='md', to='pdf',
outputfile='new_file.pdf')
# delete file text
os.remove('file.txt')

```

10.3 utiliser une feuille de style

On l'a vu, Markdown évacue complètement la question des styles (police, taille des caractères, couleur des caractères) pour ne s'occuper que de la structure du document. C'est la raison pour laquelle, beaucoup de personnes trouvent qu'il est plus simple d'écrire dans cette syntaxe. Toutefois, lorsqu'on travaille en vue de rendre un rapport ou bien lorsqu'on produit des documents formatés d'une certaine manière pour une entreprise, la production doit au final avoir une certaine forme, respecter une certaine feuille de style. Avec pandoc, il est aisé de créer un modèle adapté à ses besoins. Supposons que l'on souhaite créer un modèle de sortie en .docx (format word), il est conseillé de partir des modèles par défaut de pandoc. Pour obtenir celui qui concerne le format docx, entrer la commande suivante :

```

pandoc -o custom-reference.docx --print-default-data-file reference.docx

```

On obtient alors un document word minimal qui ne contient que des mots comme Title, heading 1, abstract. A ces entités, le document a assigné par défaut certains styles (par exemple le titre s'imprimera par défaut en Calibri 18, de couleur bleu foncé, texte 2, tandis que les titres de premier niveau (heading 1 s'imprimeront par défaut en Calibri 16, bleu accentuation 1).

Pour changer ces valeurs, il faut modifier les styles à l'intérieur de ce document, définir qu'un titre aura telle couleur, telle taille de caractères, sera dans telle police. Faire pareil pour le sous-titre, l'abstract, les titres 1, 2 et suivants, définir des marges, des entêtes, etc.

Pour convertir un document en markdown vers un document en format word qui corresponde à ce template modifié, si le template est dans le même répertoire que le document source, taper la commande suivante :

```

pandoc -o documentsource.md --reference-doc=nomdutemplate.docx -o
documentcible.docx

```

Faire de même avec un template en format open document :

```
pandoc -o custom-reference.odt --print-default-data-file reference.odt

pandoc -o documentsource.md --reference-doc=nomdutable.odt -o
documentcible.odt
```

plus d'info sur [l'aide en ligne de pandoc](#)

11. Faire un backup de ses notes sur Github

11.1 d'Obsidian à Github ou Gitlab

1. installer Git sur sa machine
2. Se créer un compte sur github
3. ouvrir github desktop
4. créer un repository (new) depuis github desktop
5. cloner ce repository n'importe où (dans un endroit où on se souviendra qu'il est), fermer le desktop de github
6. dans l'explorateur Windows, afficher les dossiers cachés
7. dans le répertoire cloné, couper le dossier .git
8. aller dans le coffre, coller le dossier .git à la racine du coffre
9. rouvrir le desktop de github. Normalement celui-ci indique qu'il ne retrouve pas le dossier .git du répertoire cloner, cliquer sur locate et sélectionner le coffre d'obsidian (sous lequel se trouve dorénavant git)
10. normalement, github desktop indexe tous les fichiers du coffre
11. faire un premier commit (entrer juste un titre de commit) et envoyer le commit
12. installer git obsidian depuis le catalogue des plugins tiers dans Obsidian
13. activer ce plugin
14. ouvrir la palette (Ctrl+P), taper Git, choisir l'option "Commit and push all changes"
15. aller sur le repository en ligne sur github, vérifier que tous les fichiers du coffre ont bien été envoyés (vérifier que l'envoi a bien été effectué)
16. définir un raccourci clavier pour l'option "commit and push all changes"

Source

On peut bien sûr envoyer le contenu de sa bibliothèque de notes et des documents qui l'accompagnent en utilisant Git directement depuis le dossier racine. Ce plugin facilite juste ce processus (un raccourci-clavier par rapport à l'envoi de quelques commandes dans le terminal de Git)

11.2 de github ou gitlab à la publication de ses notes en ligne

En théorie, il est possible de créer un [site statique](#) à partir de Github en utilisant netlify, Jekyll ou [Mkdocs](#) comme générateur de sites statique. Toutes les notes en markdown envoyées depuis Obsidian vers Github vont être disponibles depuis ce site et liées les unes aux autres par des liens hypertextes.

Après plusieurs tentatives avec Netlify, et en suivant plusieurs tutos dont [celui-ci](#), je ne suis pas encore parvenu à réaliser cette opération. A priori, elle rend inutile le fait d'obtenir ce service en publiant à ses frais ses notes sur les serveurs d'Obsidian (plugin publish d'obsidian)

12. Gérer de grands graphes de notes

voir la [méthode d'Ilya Shabanov](#) (comporte le plugin Dataview)

13. Où trouve t-on de l'aide le plus souvent à propos d'Obsidian

Quand on travaille avec Obsidian, assez fréquemment on est amené à chercher de l'information sur la syntaxe markdown, LaTeX ou Pandoc. Mais en ce qui concerne Obsidian lui-même, les deux sources d'information qui reviennent le plus souvent (hormis les tutos très nombreux et très utiles sur Youtube) sont :

- le [forum d'Obsidian](#)
- le [canal consacré à Obsidian sur Reddit](#)

Bibliographie

1. cette note de bas de page peut être placée n'importe où, elle apparaîtra forcément à la fin du document. Attention : dans la note de bas de page, ne pas utiliser les deux points (:) après les crochets.↵