



HAL
open science

Écrire autrement : réflexion croisées sur Markdown

Marine Parra, Arthur Perret

► To cite this version:

Marine Parra, Arthur Perret. Écrire autrement : réflexion croisées sur Markdown. Séminaire du Réseau Sociétés et Humanités Numériques en association avec les séminaires So MSH! et PROGEDO-Loire, Julie Aucagne, Jun 2023, Nantes, France. hal-04162643

HAL Id: hal-04162643

<https://hal.science/hal-04162643v1>

Submitted on 15 Jul 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Écrire autrement. Réflexions croisées sur Markdown

Arthur Perret (université Bordeaux Montaigne) et Marine Parra (Utrecht Universiteit)

12 juin 2023

Au programme

- C'est quoi, le format texte ?
- Réflexions sur quelques enjeux du Markdown
- Point sur le vocabulaire et présentation d'outils
- Étude de cas
- Discussion
- Atelier pratique

Présentation des intervenants

Marine Parra :

- Formation initiale en lettres -> importance du texte
- Objet de recherche : un format éditorial qui a vu le jour au XVI^e siècle -> m'a rendu sensible aux questions de mise en livre.
- Participation à un projet d'édition génétique qui m'a rendu sensible à la question des processus d'écriture.
- Difficultés avec l'exercice de rédaction de la thèse, trop de bruit, trop d'attachement à la mise en forme du texte.
- Cette année -> atelier EVEille, collaboration avec Benoît Roux

Arthur Perret :

- Formation : information scientifique, publication numérique
- Thèse : gestion des connaissances personnelles
- Thématique de recherche en lien avec la journée : outillage du travail de la connaissance (outils, formats et pratiques)
- Mon rapport à Markdown : découverte grâce à Antoine Fauchié en Master, poursuite des expérimentations durant la thèse

C'est quoi, le format texte ?

Un fichier au format texte est un fichier qui ne contient des caractères (lettres, chiffres, ponctuation, symboles, emoji...).

Voici un exemple extrait d'un fichier au format texte :

```
LE VICOMTE:  
Personne ?  
Attendez ! Je vais lui lancer un de ces traits !...
```

(Il s'avance vers Cyrano qui l'observe, et se campant devant lui d'un air fat):

Vous... vous avez un nez... heu... un nez... très grand.

CYRANO (gravement):

Très !

LE VICOMTE (riant):

Ha !

CYRANO (imperturbable):

C'est tout ?...

LE VICOMTE:

Mais...

CYRANO:

Ah ! non ! c'est un peu court, jeune homme !

On pouvait dire... Oh ! Dieu !... bien des choses en somme...

Le fait qu'un fichier soit au format texte ne signifie pas qu'il ne contienne que du texte au sens littéraire. Avec des caractères, on peut exprimer toutes sortes de choses. Voici un autre exemple :

```
Date,Open,High,Low,Close,Volume,Adj Close
2018-06-01,569.16,590.00,548.50,584.00,14077000,581.50
2018-05-01,584.90,596.76,522.18,577.73,18827900,575.26
2018-04-02,601.83,644.00,555.00,583.98,28759100,581.48
```

Date	Open	High	Low	Close	Volume	Adj Close
2018-06-01	569.16	590.00	548.50	584.00	14077000	581.50
2018-05-01	584.90	596.76	522.18	577.73	18827900	575.26
2018-04-02	601.83	644.00	555.00	583.98	28759100	581.48

Si vous utilisez régulièrement un tableur, vous aurez peut-être reconnu un fichier au format CSV (*Comma-Separated Values*). Ici, le format texte est utilisé pour exprimer des données tabu-

lares. Un autre exemple :

Un format de dessin vectoriel : SVG (*Scalable Vector Graphics*). Sert à fabriquer des images. On peut mettre du SVG dans une page web par exemple.

```
<svg style="display: none;">
  <symbol viewBox="-3 -3 30 30" id="soleil">
    <circle cx="12" cy="12" r="5"></circle>
    <line x1="12" y1="1" x2="12" y2="3"></line>
    <line x1="12" y1="21" x2="12" y2="23"></line>
```

```

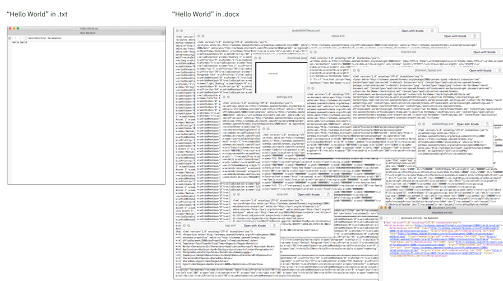
<line x1="4.22" y1="4.22" x2="5.64" y2="5.64"></line>
<line x1="18.36" y1="18.36" x2="19.78" y2="19.78"></line>
<line x1="1" y1="12" x2="3" y2="12"></line>
<line x1="21" y1="12" x2="23" y2="12"></line>
<line x1="4.22" y1="19.78" x2="5.64" y2="18.36"></line>
<line x1="18.36" y1="5.64" x2="19.78" y2="4.22"></line>
</symbol>
</svg>

```



Ces quelques exemples montrent bien la nature du format texte. Ce n'est pas un format au sens du format PDF, ou du format DOCX, ou des noms de formats cités ci-dessus (CSV, HTML, SVG). Ces formats sont spécifiques, là où le format texte est générique. On peut l'utiliser pour créer des fichiers dans toutes sortes de formats spécifiques. Le format texte est donc une sorte de méta-format. Et il est transversal à de nombreuses catégories d'objets et d'environnements logiciels.

Caractéristiques intéressantes du format texte : simple/léger ; stable/fiable ; libre (= non-propriétaire).



Comparaison format texte / format docx : simplicité. Ne veut pas dire qu'on abandonne des fonctionnalités, simplement on en déplace une grande partie vers les interfaces logicielles, et on obtient ainsi des fichiers plus légers, avec juste le contenu, lequel est facile à déchiffrer et reste donc exploitable sans

nécessiter un logiciel dédié.

Inconvénients du traitement de texte. En SHS, les logiciels de traitement de texte (comme LibreOffice Writer, Microsoft Word et Google Docs) sont utilisés par la majorité des auteurs et des éditeurs. Ce sont des outils polyvalents et puissants mais

qui comportent beaucoup d'inconvénients : leur interface est encombrée, pleine de distractions ; ils sont inutilement gourmands en ressources ; ils gèrent mal les documents longs et complexes ; ils sont fondamentalement inadaptés à la communication via le Web ; leur modèle économique est souvent défavorable à l'utilisateur ; enfin, leurs formats sont soit complètement fermés, soit difficiles à utiliser via d'autres outils.

Les logiciels basés sur du format texte constituent une alternative. En particulier, il existe des éditeurs de texte spécialisés pour l'écriture scientifique, qu'ils rendent plus efficace grâce à l'automatisation d'opérations essentielles : structurer de longs documents ; insérer des citations et générer des bibliographies ; annoter et éditer de manière collaborative ; exporter les textes dans les formats requis par les éditeurs.

Réflexions croisées sur Markdown

1. Simplifier son processus de travail
2. Quelles interfaces ?
3. Publier sur le web

Avant d'entrer dans le vif du sujet : pourquoi parler de Markdown ?



– **Légèreté.** Balisage dit léger permet de structurer simplement et rapidement un document pour un rendu intelligible à la fois pour l'homme et pour la machine

– **Simplicité.** Utilisation économe de balises composées sur des caractères de ponctuation familiers à toute personne ayant déjà approché un clavier

d'ordinateur (#, -, *, =)

– **Autonomie.** Liberté d'outils (dans l'absolu, fonctionne avec n'importe quel éditeur de texte brut)

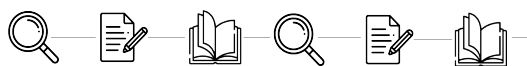
Il est par conséquent simple à maîtriser et libère l'auteur·rice de son outil. Une ressource possible dans une démarche *low tech*.

1. Simplifier son processus de travail

En théorie : on cherche, on écrit, on publie.



Sur le temps long, cela devient cyclique : on cherche à nouveau, on écrit à nouveau, on publie à nouveau.



Mais en réalité : on prend aussi des notes sur papier, on part de textes publiés dans différents formats, on réutilise des communications orales, etc.



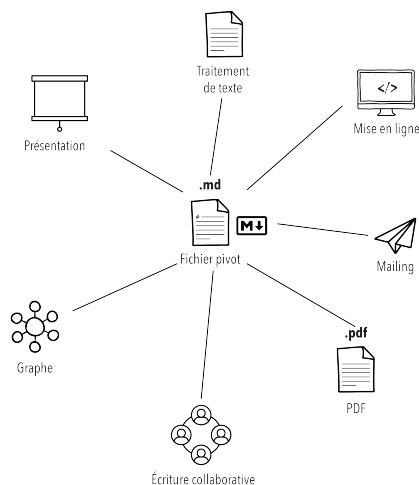
Les outils d'écriture sont souvent plus adaptés à un processus de travail "théorique" qu'à la réalité de nos **méthodes** et de nos **gestes** : dupliquer, copier-coller, exporter...



L'exemple du copier-coller : beaucoup de textes sont lourds, peu maniables, difficiles à interroger et à convertir. Le formatage, recommencé par chaque

instance de la chaîne de partage des connaissances (auteur·rice·s, co-auteur·rice·s, reviewers, éditeur·rice·s, imprimeur·e·s), constitue une débauche de temps et d'énergie. Or on le perd souvent lors du copier-coller (notamment les caractères invisibles).

Simplifier son processus de travail, c'est aussi **constituer un environnement de travail**. De quoi a-t-on besoin ?



Marine Faou, Bécot Roux, schéma d'un environnement de travail

- **Circuler** de façon fluide dans sa production écrite
- **Mobiliser** différents morceaux de texte dans différents contextes de recherche
- Minimiser le **temps de traitement**
- Minimiser le temps de **mise en forme**
- Constituer un **écosystème de travail** global

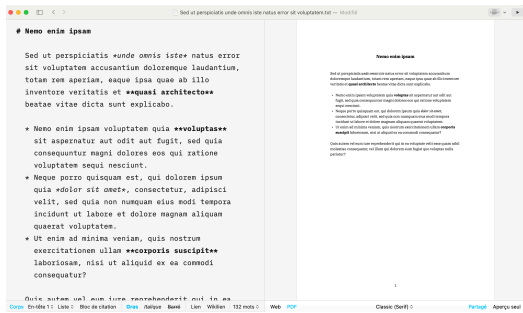
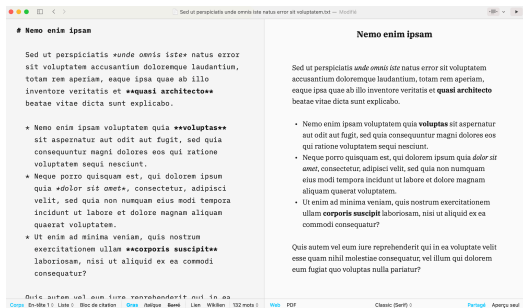
Tout ceci répond aux enjeux de la **maniabilité** du texte, un besoin incompatible avec la variété des formats, pluralité d'outils, etc.

La maniabilité des textes en Markdown permet de circuler de façon plus fluide dans sa production écrite et de la mobiliser au fil de différents contextes de recherche (communications scientifiques, articles de recherche, notes, supports de cours, etc.). Le temps de traitement et le travail de mise en forme des textes ainsi produits sont de ce fait rationalisés. L'ensemble de ces étapes constitue un écosystème de travail global et intégré.

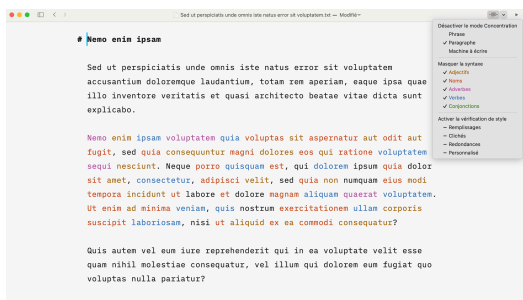
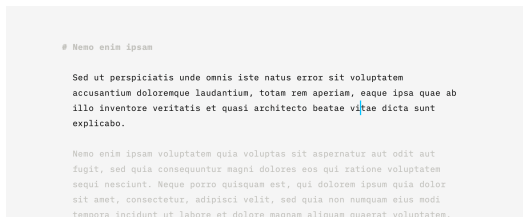
2. Interfaces ergonomiques

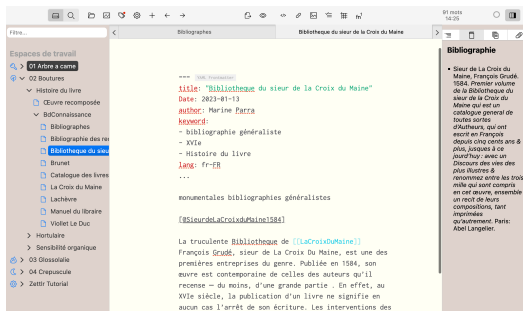
Il existe de nombreux éditeurs de texte spécialisés dans l'écriture en Markdown.

Ces éditeurs proposent souvent une distinction entre fond et forme, avec une visualisation du texte au format HTML ou PDF.



Les interfaces peuvent aller d'une grande simplicité à des environnements beaucoup plus chargés en fonctionnalités (ici l'exemple d'iA Writer puis de Zettlr).





L'ergonomie de ces interfaces favorise la concentration. Pour les logiciels spécialisés (comme Zettlr dans le cas de l'écriture scientifique), l'ergonomie est aussi pensée en fonction de formes d'écriture particulières (dans Zettlr, l'insertion des citations est grandement simplifiée par rapport à un traitement

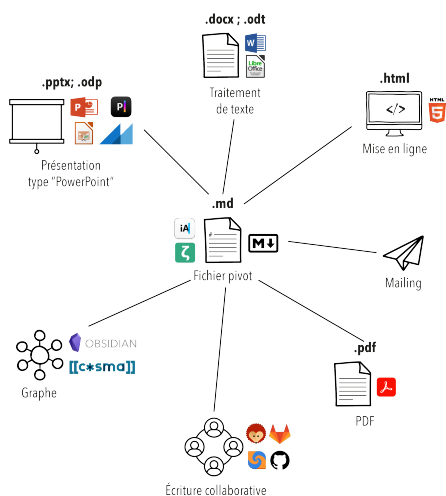
de texte).

3. Publier sur le web

Markdown a été pensé pour le Web : c'est une sorte de « sténographie » de HTML. Contrairement aux logiciels de traitement de texte, un environnement d'écriture basé sur Markdown facilitera donc intrinsèquement le passage à la publication web. Certains outils le permettent de manière intégrée, comme par exemple l'envoi du texte vers WordPress depuis l'éditeur de texte.

Vocabulaire & présentation d'outils

Pour chaque tâche, plusieurs outils basés sur Markdown sont envisageables, plusieurs configurations, donc plusieurs environnements de travail.



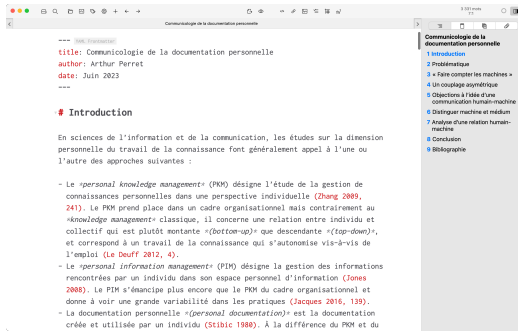
Maïne Paria et Benoît Rioux, schéma d'un environnement de travail (2)

Rédiger un article en Markdown

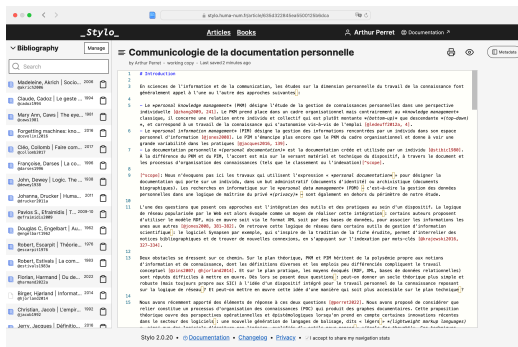
Rédiger un article signifie un certain nombre de besoins :

- d'écriture : citations, notes, tableaux, figures, références croisées, formules mathématiques, etc.
- d'interfaces pour le suivi éditorial, potentiellement de manière collaborative (re-lecture, corrections) : annotation, comparaison entre versions...
- de modèles : styles, métadonnées.

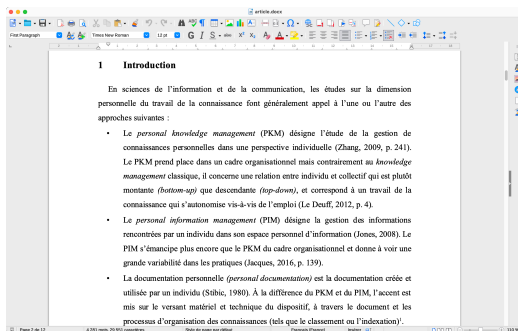
Nous avons déjà mentionné **Zettlr**, un éditeur de texte basé sur Markdown et spécialisé dans l'écriture scientifique.



Un exemple d'outil qui intègre toutes les étapes de la chaîne éditoriale : **Stylo** (disponible via la plateforme Huma-Num). Comme Zettlr, Stylo est basé sur Markdown et Pandoc (convertisseur permettant de transformer le texte en Markdown en d'autres formats). L'outil évolue régulièrement, avec récemment l'intégration de références à partir d'une bibliothèque Zotero, et l'insertion des citations via un raccourci à la Zettlr.



Ces outils permettent d'exporter un texte dans différents formats, par exemple pour traitement de texte, afin de collaborer avec d'autres personnes (auteurs, relecteurs, éditeurs...).



Créer un carnet de recherche en Markdown

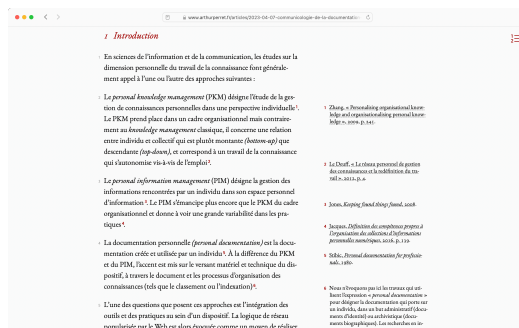
Markdown est une modalité possible d'écriture pour les carnets de recherche (ou blogs scientifiques). Intérêt : conserver le même format d'écriture quel que soit le format de publication. Ainsi on écrit pareil son article dans Stylo pour une revue que pour un blog.

Autre intérêt : on peut changer d'outil de blog s'il ne nous convient pas. Les données (textes, billets de blog) peuvent être "déménagés" tels quels d'un outil à l'autre.

Dans certains domaines, il y a des environnements d'écriture qui intègrent la création de blogs/sites web. Exemple : en statistiques et science des données, l'écosystème R (avec R Markdown, R Studio, Quarto etc.) permet d'écrire des articles et de créer des sites web, le tout à partir du même format Markdown.

Au-delà, il existe des outils pour créer des sites web entièrement à la main à partir de fichiers Markdown, comme Hugo. Et pour des usages très simples, plus « manuels » et *low-tech*, on peut utiliser le convertisseur Pandoc en tandem avec un programme d'automatisation comme Make, pour générer un site à partir de fichiers Markdown.

(En capture d'écran : le site web d'Arthur Perret.)



Créer une présentation en Markdown

Plusieurs logiciels de présentation basés sur Markdown existent. Comme les éditeurs de texte, ils proposent de séparer le fond (le fichier texte) et la forme (le rendu final de la présentation, mise en forme).

On peut citer Marp, ainsi que iA Presenter (outil utilisé pour cette présentation).

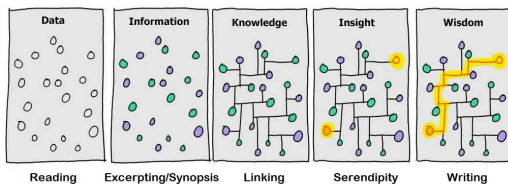


Le principe de base est simple : on rédige en Markdown et on ajoute le séparateur `--` pour signaler la fin d'une diapo et le début de la suivante.

iA Presenter se distingue par ses choix d'ergonomie. Contrairement aux logiciels type PowerPoint ou Keynote, on rédige un texte qui ne s'affichera pas sur les diapos, et on choisit d'afficher certaines choses (titres, images, paragraphe).

Une nouvelle génération d'outils de prise de notes a vu le jour ces dernières années, autour de la logique des wikis : mettre en relation ses notes. La plupart sont des logiciels basés sur Markdown. Pour ces outils, on parle beaucoup de « méthode Zettelkasten », il s'agit en fait une nouvelle incarnation d'une pratique universitaire très traditionnelle, celle de la fiche érudite, qui descend en droite ligne des livres de lieux communs. Tout cela remonte au moins au XVe siècle, mais la plupart des personnes qui s'enthousiasment pour la gestion des connaissances personnelles ne s'intéressent pas beaucoup aux origines du domaine et ne lisent pas la littérature scientifique, d'où un discours très orienté vers la « nouveauté ».

À la façon d'un wiki personnel, cette documentation regroupe tous les documents dans lesquels on travaille : fiches de lecture, notes terminologiques, brouillons d'idées, etc. C'est l'*espace de travail* dans Zettlr ou le "*vault*" dans Obsidian. L'idée centrale est de travailler avec des notes organisées de manière non-linéaire, qui se font référence les unes aux autres. Périodiquement, une idée émerge : un lien nouveau entre deux choses (ou plus). Les notes servent d'aide-mémoire et d'espace de réflexion/idéation.



Source de l'illustration : [Chris Aldrich](#) .

Il y a de nombreuses manières de faire ça techniquement. Exemple : base de données relationnelles. Mais une base de données n'est pas forcément indis-

pensable : on peut se contenter de fichiers au format texte. Beaucoup d'outils proposent ça, en Markdown. Les liens entre fichiers se font avec la syntaxe des wikis.

Exemple dans Zettlr :

```

---
title: Everything bucket
id: 20210721113916
type: concept
---

Un «everything bucket» est un
[[20200625120311]] dispositif
utilisé pour enregistrer et
organiser n'importe quel type de
[[20200807144111]] données.

Le concept est proposé et critiqué
par Alex Payne dans "The Case
Against Everything Buckets" (2002).
Il est défendu par David Karger dans
"Why All Your Data Should Live in
One Application" (2010).

---
title: Dispositif
id: 20200625120311
type: concept
---

Objectif du dispositif :

> « Le dispositif est destiné à
maîtriser l'incertitude du
devenir » (Bachimont 2004, 17).

À rapprocher de la réduction de la
complexité, chez Luhmann
("Observations on modernity", 1998,
pp. 6-7) et chez Robert dans
«Mnēotechnologies» (Robert 2010),
ainsi que de la réduction de
l'incertitude chez Buckland ;

```

```

---
title: Everything bucket
id: 20210721113916
type: concept
---

Un «everything bucket» est un
[[20200625120311]] dispositif
utilisé pour enregistrer et
organiser n'importe quel type de
[[20200807144111]] données.

Le concept est proposé et critiqué
par Alex Payne dans "The Case
Against Everything Buckets" (2002).
Il est défendu par David Karger dans
"Why All Your Data Should Live in
One Application" (2010).

---
title: Dispositif
id: 20200625120311
type: concept
---

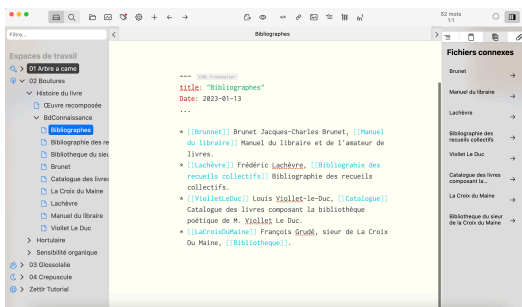
Objectif du dispositif :

> « Le dispositif est destiné à
maîtriser l'incertitude du
devenir » (Bachimont 2004, 17).

À rapprocher de la réduction de la
complexité, chez Luhmann
("Observations on modernity", 1998,
pp. 6-7) et chez Robert dans
«Mnēotechnologies» (Robert 2010),
ainsi que de la réduction de
l'incertitude chez Buckland ;

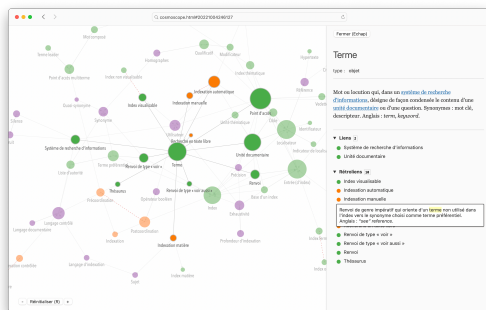
```

Exemple d'une base de connaissance personnelle consacrée à une recherche sur des bibliophages :

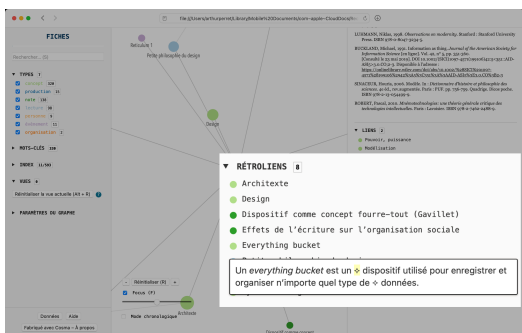


Chaque fichier (ou « fiche », si on reprend cette terminologie ancienne mais toujours pertinente) porte sur une chose : une personne, une œuvre, un concept... Comme dans une base de données, on applique des méthodes d'organisation des connaissances : des catégories, des étiquettes, etc.

Une base de connaissance peut vite devenir un fouillis difficile à naviguer. La visualisation présente alors un intérêt. Cosma est un logiciel utilisable en tandem avec Zettlr qui permet de visualiser, naviguer et partager une base de connaissances faite de fichiers Markdown interreliés. Le logiciel a été imaginé par Arthur Perret ; il est développé par Guillaume Brioude.



Contrairement au Web, dans lesquels les liens sont à sens unique, ici chaque fichier est visualisé avec ses « rétroliens » contextualisés : on peut savoir où un fichier a été mentionné ailleurs dans la base, et de quelle manière.



Il existe des outils intégrés permettant de constituer ce genre de base de connaissances de façon collective (ex : Roam, Notion). Mais ces outils sont généralement propriétaires, leur code est fermé, ils sont accessibles uniquement en ligne ; tout ceci peut susciter des inquiétudes sur la pérennité de l'accès aux données, alors qu'une base de connaissances est censée être utile sur le long terme.

Une approche plus autonome et robuste est l'utilisation d'un dossier partagé via un logiciel de synchronisation et/ou de versionnement. Exemple : Git, notamment via le service GitLab (disponible via la plateforme Huma-Num).

Conclusion(s)

Markdown a des points faibles. Il est peu expressif, ce qui nécessite d'utiliser une variante adaptée à son usage, comme le Pandoc Markdown pour l'écriture scientifique, ou le R Markdown pour les statisticiens, etc. Chacun doit trouver son environnement de travail, et nous avons présenté quelques options parmi une multitude d'autres qui existe, ce qui peut dérouter.

Nos choix d'écriture correspondent à de vrais besoins, que nous n'avons pas toujours le temps de préciser ou conscientiser, parce que ce sont en fait des gestes, des habitudes héritées mais rarement interrogées. Très souvent, quand on demande à des étudiants ou des collègues pourquoi ils écrivent sur Word, ils sont incapables de véritablement l'expliquer. L'enjeu pour nous est de choisir soigneusement les outils avec lesquels nous travaillons, ce qui ne peut se faire que dans une démarche réflexive.