# Documentation Stage Gaëlle Cosma/stylo

## Contexte

Cosma est un outil de visualisation conçu pour les professionnels de l’information qui transforme des fichiers de texte simple avec des liens wiki en un réseau dynamique de cartes indexées. Cosma est prévu pour se superposer à un éditeur de texte tel que Stylo ; comme tous deux utilisent le langage markdown et héritent d’une philosophie des formats ouverts et de l’interopérabilité, leur articulation est donc non seulement possible, mais entrevue par les développeurs de Cosma. L’enrichissement et l’exploitation des données sémantiques contenues dans les articles est également une des pistes de développement en cours pour l’éditeur Stylo. Cette intégration permet un enrichissement des données des articles sur Stylo, pour les auteurs comme pour les revues. Les auteurs pourront valoriser les liens entre leurs propres textes, dans un prolongement de leurs propres activités de recherche en sciences humaines, et analyser et synthétiser leurs données de recherche de façon innovante et opérante. Les revues, quant à elles, pourront visualiser les références, les enjeux, les thématiques traversant l’ensemble de leur production, mais aussi des unités plus discrètes, comme les dossiers ou les articles eux-mêmes.

L’objectif serait de pouvoir visualiser le cosma en sélectionnant les articles. Une visualisation avec plusieurs articles : pour chaque article on peut visualiser le nœud qui est l’article, et les références bibliographiques connectées à l’article, et des liens avec des articles qui partageraient les mêmes références. Cela permettrait également de créer des liens dans sa bibliographie et de voir les différents liens de bibliographie entre les articles. L’intérêt premier est de pouvoir voir s’il y a des références mises à part ou, au contraire, des références qui lient plusieurs articles.

## Mofification pour généré le cosma pour une revue ou un corpus.

### En tête YAML

Sur les fiches Markdown, nous avons récupéré tout le texte en markdown, mais nous avons ajouté une en-tête YAML à la main : Exemple :

---  
title : De \*Lourdes\* à \*Paris\*, en train ou à bicyclette <!-- peut être récupéré avec les métadonnées dans stylo. -->  
types: <!-- pas obligatoire, peut être rajouté aux métadonnées dans stylo et récupéré ou rajouté par défaut. Nous avons pensé qu'il serait plus pertinent de créer des types tel que "article", "undefined" ou "références". -->  
 - article  
tags: <!-- pas obligatoire, peut être récupéré avec les métadonnées dans stylo ("mots clés" dans stylo) -->  
 - hybridations médiatiques   
 - machiniques dans le cycle des \*Trois Villes\*   
 - emile Zola (1893-1898)  
---

### Bibliographie

Nous avons créé un dossier avec :

* Roch a créé un fichier contenant des métadonnées décrivant les références bibliographiques qui regroupé toutes les bibliographies cités dans toutes les fiches et non un fichier de bibliographie par fiches. Le format requis est CSL JSON (extension .json).
* Nous avons ajouté un fichier contenant les règles de formatage pour les citations et les bibliographies. Le format requis est CSL (extension .csl), que nous avons téléchargé à partir du répertoire des styles CSL de Zotero.
* Roch ajouté un fichier contenant les termes localisés utilisés dans les bibliographies (par exemple « éditeur », « numéro »…). Le format requis est XML (extension .xml).

Exemple de format JSON :

{  
 "author": [  
 {  
 "family": "Huret",  
 "given": "Jules"  
 }  
 ],  
 "id": "huret\_enquete\_1891",  
 "issued": {  
 "date-parts": [  
 [  
 1981  
 ]  
 ]  
 },  
 "publisher": "Charpentier",  
 "publisher-place": "Paris",  
 "title": "Enquête sur l’évolution littéraire",  
 "type": "book"  
 },

### CSS

Nous avons ajouté un CSS en plus, mais ce n’est pas obligatoire.

### Fichier config

Nous avons généré le config.yml avec la commande Cosma puis modifié le YAML.

select\_origin: directory files\_origin: ./fiches nodes\_origin: “” links\_origin: “” nodes\_online: “” links\_online: “” images\_origin: “” export\_target: “” history: true focus\_max: 2 record\_types: undefined: fill: “#005C53” stroke: “#005C53” litterature: fill: “#DBF227” stroke: “#DBF227” test: fill: “#D6D58E” stroke: “#D6D58E” Concept: fill: “#042940” stroke: “#042940” perso: fill: “#005C53” stroke: “#005C53” référence: stroke: “#6C6C6C” fill: “#6C6C6C” link\_types: undefined: stroke: simple color: “#e1e1e1” references\_as\_nodes: true references\_type\_label: “référence” record\_filters: [] graph\_background\_color: “#ffffff” graph\_highlight\_color: “#ff6a6a” graph\_highlight\_on\_hover: true graph\_text\_size: 10 graph\_arrows: true node\_size\_method: degree node\_size: 10 node\_size\_max: 20 node\_size\_min: 2 attraction\_force: 200 attraction\_distance\_max: 250 attraction\_vertical: 0 attraction\_horizontal: 0 views: {} record\_metas: [] generate\_id: always link\_context: tooltip hide\_id\_from\_record\_header: false title: “” author: “” description: “” keywords: [] link\_symbol: “” csl: ./bibliographie/apa.csl bibliography: ./bibliographie/biblio.json csl\_locale: ./bibliographie/locales-fr-FR.xml css\_custom: ./csstest.css devtools: false lang: en

Questions (Avec Antoine):

quel format a la bibliographie au format JSON ? Ou plutôt : comment le fichier JSON est produit ?  
quel identifiant utiliser pour éviter les doublons ? Actuellement les références sont identifiées avec les clés bibtex attribuées par l'API Zotero ;  
que faire des types ou des mots-clés ? Idée : attribuer des mots-clés (par exemple thématiques) aux références pour voir des (nouveaux) liens entre des références différentes entre différents articles ;

**comment gérer l’affichage de l’article dans Cosma ? Soit avoir une option de déplier ou qu’une partie. Surtout que les informations sur la biblio sont à la fin l’intérêt est tout de même plus pour visualiser rapidement des connexions thématiques ou commerciales**

## Idées pour l’implémentation Cosma

### Revue

* Lier les intervenants, mots-clés, notions, termes entre eux dans le même dossier.
* En quoi ces auteurs appartiennent-ils, qui sont-ils ? [style de Open Data Sphere].
* Quelque chose dans un article peut avoir un lien avec un autre sans s’en rendre compte.

### Général

* Mieux comprendre sa pensée et sa structure :
  + Personnes, concepts, Wikidata.
  + Réseau de personnes.
* Explorer l’idée de pouvoir visualiser différentes versions et différences, tel que mon rapport.
* Citations.

## Besoins spécifiques de Cosma

* Voir doc.
* Possibilités avec Wikidata ?

### CSS

Nous avons ajouté un CSS en plus, mais ce n’est pas obligatoire.

Exemple de CSS :

/\*Feuille de style réalisé à l'occasion du projet d'open data sphère à l'iut Bordeaux Montaigne en 2024 \*/  
  
  
@import url('https://fonts.googleapis.com/css2?family=Bitter:ital,wght@0,100;0,300;0,400;0,500;0,700;1,400;1,500;1,700&family=Space+Mono:wght@400;700&display=swap');  
  
.record-title {  
 font-family: 'young serif', monospace;  
 font-size: 23px;  
 color: #CE6C6C;  
}  
  
p {  
 font-family: 'Times New Roman', Times, serif; /\* Police classique souvent utilisée dans les documents universitaires \*/  
 font-size: 16px; /\* Taille de police standard pour les documents imprimés \*/  
 line-height: 1.6; /\* Espacement des lignes pour une lecture confortable \*/  
 color: #333; /\* Couleur de texte foncée pour une meilleure lisibilité \*/  
 background-color: #f8f8f8; /\* Fond légèrement gris pour imiter le papier \*/  
 padding: 15px; /\* Espacement autour du texte pour le démarquer \*/  
   
 margin: 20px 0; /\* Marges pour séparer les paragraphes et imiter le style d'un document imprimé \*/  
 text-align: justify; /\* Justification du texte pour un aspect plus formel \*/  
 hyphens: auto; /\* Césure automatique des mots pour une meilleure mise en page \*/  
}  
  
  
button {  
 background-color: #CE6C6C;  
 color: white;  
 border-radius: 6px;  
 border: none;  
 padding: 5px 10px;  
 transition: all 0.1s ease;  
}  
  
button:hover {  
 animation-duration: .8s;  
 animation-name: clignoter;  
 animation-iteration-count: infinite;  
 transition: none;  
}  
  
.btn {  
 background-color: #CE6C6C;  
 border: 1px solid var(--border-color);  
 border-radius: 2px;  
}  
h1.title {  
 font-family: 'Space Mono', monospace;  
 text-transform: uppercase;  
 font-size: 27px;  
 color:#CE6C6C;  
 text-align: center;  
}  
  
aside.menu header {  
 display: flex;  
 align-items: center;  
 justify-content: center;  
}  
  
aside.menu header:before {  
 display: inline-block;  
 height: 70px;  
 width: 100px;  
 background-size: 100px 70px;  
 background-repeat: no-repeat;  
}  
article.record header {  
 display: block;  
 gap: 0 1em;  
 grid-template:  
 "thumbnail title" auto  
 "thumbnail nom" auto  
 "thumbnail prénom" auto  
 "thumbnail type" auto  
 "tags tags" auto;  
}  
  
h1.record-title {  
 grid-area: title;  
}  
  
.record header img {  
 height: auto;  
 max-width: 50%;  
 margin: 1rem 0;  
}  
.record img {  
 height: auto;  
 max-width: 50%;  
}  
.record-thumbnail span:nth-child(2) {  
 grid-area: thumbnail;  
 display: inline-block;  
 width: 100px;  
 height: 100px;  
 background-image: url(attr(content));  
 background-size: cover;  
 background-repeat: no-repeat;  
 content: "";  
}  
  
div.record-id {  
 display: none;  
}  
div.record-nom {  
 grid-area: nom;  
}  
  
div.record-prenom {  
 grid-area: prénom;  
}  
  
div.record-type {  
 grid-area: type;  
}  
div.record-nom,  
div.record-prenom {  
 width: 50%; /\* Ajustez cette valeur si nécessaire \*/  
}  
  
  
div.record-tags {  
 grid-area: tags;  
 grid-column: 1 / 3; /\* Cela permet aux tags de s'étendre sur les deux colonnes de la grille \*/  
}  
  
div.record-tags button.btn{  
 border-radius: 40px;  
 background-color: transparent;  
 border-color: #CE6C6C;  
 color :#CE6C6C;  
}