

Knowledge & Knowledge-clustering

Exposé mensuel *GU*Tenberg
5 décembre 2024

Thomas Colcombet



knowledge

knowledge est

- un paquet pour LaTeX (écrit en LaTeX3)
- Dont le développement a commencé vers 2014
- accessible sur CTAN depuis 2017 (<https://ctan.org/pkg/knowledge>)

Objectif : améliorer l'accès à l'information pour le lecteur d'un document [scientifique], en particulier en version électronique, et faciliter le travail de l'auteur dans cette tâche.

Le souhait du lecteur : ne pas avoir à tout lire, et être capable de retrouver toutes les informations importantes facilement. Cela signifie :

- avoir un bon index dans une version papier
- pouvoir retrouver les notions en un click dans une version électronique (PDF / HTML).

Le souhait de l'auteur : ne pas avoir à perdre de temps dans la rédaction, et ne pas être interrompu dans le processus.

Exemples d'utilisation

- Créer des liens entre un terme et sa définition.
- Maintenir un index.
- Gérer les liens dans une page web
exemple: <https://www.irif.fr/~colcombe>, rédigé avec htlatex et knowledge pour les liens.

Exemples

A. Casares, **Propriétés structurelles des automates sur les mots infinis et mémoire pour les jeux**, Thèse de doctorat de l'université de Bordeaux (23/11/2023) [<https://antonio-casares.github.io/Publications/Casares23-PhDThesis.pdf>]

C. Mascle, **Vérification et synthèse de systèmes distribués à synchronisation faible**, Thèse de doctorat, université de Bordeaux (28/11/2024) [<https://corto-mascle.github.io/papers/Manuscrit.pdf>].

T. Colcombet, S. v. Gool et R. Morvan, **First-order separation over countable ordinals**, arXiv 3 (2024) [<https://arxiv.org/abs/2201.03089>].

Plan

- Premier contact
- Définitions et utilisations : notions
- Knowledge-clustering
- Gestion d'un index
- Mathématiques

Premier contact

Préambule

```
\documentclass{article}
```

```
\usepackage[hidelinks,breaklinks]{hyperref}
```

```
\usepackage{xcolor}
```

```
\usepackage{knowledge}
```

```
\knowledgeconfigure{quotation,notion}
```

```
% Série de commandes \knowledge, ou:
```

```
\input{main-knowledge.kl}
```

```
\begin{document}
```

```
  [contenu du document]
```

```
\end{document}
```

Paquets souvent chargés **avant** knowledge et qui activent des fonctionnalités.

Options de configuration pratiques.

Il est bien (mais pas obligatoire) d'importer les commandes « \knowledge » depuis un fichier séparé, en « .kl » .

Trois modes d'activation

```
\usepackage{knowledge}
```

Par défaut le mode « composition » rend les erreurs et warnings visibles.

À garder pendant la rédaction.

```
\usepackage[electronic]{knowledge}
```

Le mode « electronic » produit une version finale « colorée » pour rendre les liens visibles au lecteur.

À utiliser pour la production finale d'un document prévu pour être consulté sur un support numérique.

```
\usepackage[paper]{knowledge}
```

Le mode « paper » produit une version finale en noir et blanc, avec les liens.

À utiliser pour la production finale du document à imprimer.

Utiliser une connaissance

Certains termes sont utilisés avec `\kl{...}` ou "...".

Nécessite l'option
« quotation »

Le Groupe francophone des Utilisateurs de TeX ("GUT") est une "association loi 1901", fondée en 1988. Il a pour but de promouvoir l'utilisation de "TeX" et "LaTeX" en offrant à ses adhérents un ensemble services autour de ces logiciels, aidant à leur découverte et à leur utilisation.

main.tex

pdflatex

Le Groupe francophone des Utilisateurs de TeX (GUT) at été fondé est une association loi 1901, fondée en 1988. Il a pour but de promouvoir l'utilisation de TeX et LaTeX en offrant à ses adhérents un ensemble services autour de ces logiciels, aidant à leur découverte et à leur utilisation.

main.pdf

En mode « composition », les termes inconnus apparaissent en orange.

* Summary *

4 undefined knowledge(s).

0 autoreference(s) are introduced

0 autoreference(s) are used by

0 autoreference(s) are properly

0 autoreference(s) are defined

* Undefined knowledges *

```
\knowledge{ignore}
```

```
| GUT
```

```
| association loi 1901
```

```
| TeX
```

```
| LaTeX
```

main.diagnose

Associer des comportements : `\knowledge`

`\knowledge` permet d'associer un comportement à un terme:

Syntaxe 1

```
\knowledge{TeX}{text=\TeX, url=https://fr.wikipedia.org/wiki/TeX}
```

```
\knowledge{LaTeX}{text=\LaTeX, url=https://fr.wikipedia.org/wiki/LaTeX}
```

Syntaxe 2 (notation-barre, ne peut pas être utilisée dans une macro)

```
\knowledge{color = blue, url=https://www.gutenberg-asso.fr/}
```

```
| GUT
```

```
\knowledge{url = https://fr.wikipedia.org/wiki/Association_loi_de_1901 }
```

```
| association loi 1901
```

```
| loi 1901
```

} 1 « synonyme » par ligne

pdf~~l~~atex

Liens cliquables

Le Groupe francophone des Utilisateurs de TeX (**GUT**) est une association loi 1901, fondée en 1988. Il a pour but de promouvoir l'utilisation de TeX et L^AT_EX en offrant à ses adhérents un ensemble services autour de ces logiciels, aidant à leur découverte et à leur utilisation.

Bonne pratique : mettre ces commandes dans un fichier séparé « .kl » .

Directives utilisables par `\knowledge`

Quelques directives préexistantes :

`boldface`

`color={color as for xcolor}`

`emphasize`

`ensuretext`

`ensuremath`

`italic`

`url={url}`

`typewriter`

`underline`

`ref={label latex}`

`wrap= \formatting_macro`

etc...

Créer une nouvelle directive avec `\knowledgedirective`

```
\knowledgedirective{wikipedia}{color=blue,  
url={https://fr.wikipedia.org/wiki/#1}}
```

```
\knowledge{wikipedia=Groupe_(mathématiques)}  
| groupe  
| groupes
```

Remplacement et portée

« texte » affiché

la « portée » : où chercher le terme

le « terme »
définissant le
comportement

"texte@terme@portée"

\kl[terme](portée){texte}

"texte"

"texte@@portée"

\kl{texte}

\tkl(portée){texte}

"texte@terme"

"texte@terme@portée"

\kl[terme]{texte}

\kl[terme](portée){texte}

Les "groupes" sont des objets algébriques importants.

"Ils@groupe" sont équipés de "morphismes@@grp".

A le même comportement que **groupe** (e.g. pointe sur la même ressource).

Réfère à un **morphisme de groupe** (différent d'un morphisme de corps).

```
\knowledge{wikipedia=Morphisme_de_groupes}
```

```
| morphisme de groupe
```

```
| morphisme@grp
```

```
| morphismes@grp
```

« @ » dans le notation-barre spécifie la portée de la définition

Définitions et utilisations : notions

Notion (autoref)

Objectif : relier l'usage d'une notion par un hyperlien vers le lieu de son introduction.

```
\knowledgeconfigure{notion}
```

configure automatiquement un mécanisme,
et définit la directive « **notion** ».

```
\knowledge{notion}
```

```
| monoïde  
| monoïdes
```

```
\knowledge{notion}
```

```
| groupe  
| groupes
```

La directive « **notion** » :

- permet d'introduire une fois, et
- d'utiliser autant de fois que l'on veut un terme
- les utilisations pointent alors vers l'introduction
- un aspect adapté est choisi (suivant le mode)

Un **"monoïde"** est un ensemble équipé d'une opération binaire associative et d'un élément neutre.

Un **"groupe"** est un **"monoïde"** équipé d'une opération d'inverse.

Les notations `\intro{terme}`
et **"terme"** permettent de préciser qu'il
s'agit du lieu de définition du concept.

Points d'ancrage : `\AP` et `\itemAP`

Problème : Dans le HTML produit, les liens font revenir au point de « début de block » le plus récent (début de section, définition, ...) qui peut être plusieurs pages avant.

La macro `\AP` introduit un « point de destination » quelques millimètres à gauche de la zone de texte, au niveau de l'appel, et qui sert de destination pour les notions définies après (similaire à `\phantomsection`).

`\AP` Un `"monoïde"` est un ensemble équipé d'une opération binaire associative et d'un élément neutre.
`\AP` Un `"groupe"` est un `"monoïde"` équipé d'une opération d'inverse. `\AP`
Un `"morphisme de groupe"` est une application entre deux `"groupes"` qui préserve les opérations de `"groupe"`.

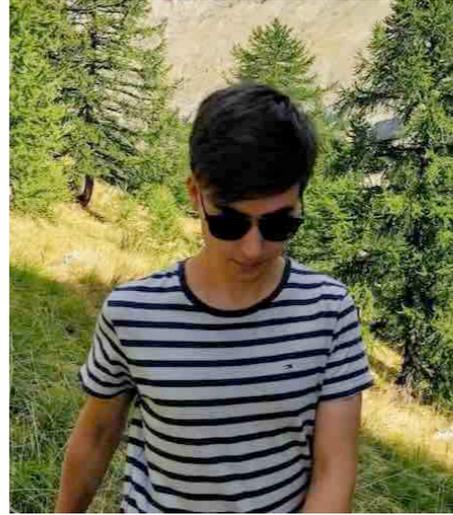
- ▮ Un *monoïde* est un ensemble \mathcal{M} muni d'un élément neutre e et d'une opération binaire associative.
- ▮ Un *groupe* est un *monoïde* muni d'une opération d'inverse.
- ▮ Un *morphisme de groupe* est une application entre deux groupes qui préserve les opérations de groupe.

En mode « composition », les points d'ancrage sont indiqués par un angle.

La macro `\itemAP` combine `\item` et `\AP` (`\AP` ne peut pas être utilisé dans l'argument facultatif de `\item`).

Knowledge-clustering

knowledge-clustering



Knowledge-clustering est un outil en ligne de commande écrit en python

- par Rémi Morvan (<https://www.morvan.xyz/>)
- présent sur github (<https://github.com/remimorvan/knowledge-clustering>)
- installation : `python3 -m pip install --upgrade knowledge-clustering`
- qui possède trois fonctionnalités :
 - cluster** permet de regrouper les termes similaires automatiquement
 - addquotes** essaye d'inférer dans le corps du texte où des guillemets pourraient être ajoutés
 - anchor** permet d'identifier où pourraient manquer des points d'ancrage

knowledge cluster

permet de grandement simplifier l'écriture des commandes \knowledge :

```
> knowledge cluster -d main_file.diagnose -k main_knowledge_file.kl
```

ou plus simplement si aucune ambiguïté :

```
> knowledge cluster
```

Knowledge-clustering va alors :

- collecter les termes non-encore définis depuis le fichier « [.diagnose](#) »
- voir s'ils sont proches de termes déjà définis dans une commande \knowledge dans le fichier « [.kl](#) »
- si c'est le cas, va proposer leurs ajouts au fichier « [.kl](#) » comme des lignes de commentaire
- puis les termes restants sont regroupés en paquets similaires (cluster)
- des lignes de commentaires correspondantes sont ajoutées au fichier « [.kl](#) »

Gestion d'un index

Index

Charger `makeidx` ou `imakeidx`
avant `knowledge`

```
\usepackage{makeidx}  
\makeindex  
\usepackage{knowledge}  
  
[...]  
  
\printindex
```

Indiquer où mettre les notions dans l'index :

```
\knowledge{notion, index=monoïde}  
| monoïde  
| monoïdes
```

```
\knowledge{notion, index=groupe}  
| groupe  
| groupes
```

```
\knowledge{notion, index=morphisme,  
index parent key=groupe}  
| morphisme de groupe  
| morphismes de groupe  
| morphisme@groupe  
| morphismes@groupe
```

`index={aspect dans l'index}`

`index key={clef pour placer dans l'index}`

`index parent key={clef parente si dans un sous-index}`

Mathématiques

Méthode 1 : comme en mode texte

Il est pratique/conseillé d'utiliser comme clef pour knowledge la fonction elle-même.

```
\newcommand\successor{\kl[\successor]{\mathit{succ}}}  
\knowledge{\successor}{notion}
```

Pour l'introduction, il faut utiliser un modificateur : `\intro*` déclare que le prochain `\kl` doit se comporter comme un `\intro`.

```
\AP We define  $\text{\intro*}\text{successor}(n)$  colon  $\mathbb{N}$  to  $\mathbb{N}$   
by  $\text{\intro*}\text{successor}(n)=n+1$ .
```

⌈ We define *succ*(*n*): $\mathbb{N} \rightarrow \mathbb{N}$ by *succ*(*n*) = *n* + 1.

`\reintro*` donne l'aspect de `\intro`, mais sans le lien.

`\phantomintro`{\successor} permet de faire un lien, mais sans afficher

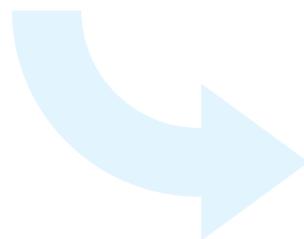
Méthode 2 : `\knowledgenewcommand`

`\knowledgenewcommand` combine `\knowledge{notion}` et `\newcommand` :

```
\knowledgenewcommand\bracket[1]{\cmdkl{[]#1\cmdkl{[]}}
```

`\cmdkl` remplace alors `\kl`. Il peut être utilisé plusieurs fois dans la macro.

`\AP` We define `\intro*` `\bracket a` to be the equivalence class of `~ $a`.
This is a congruence: `\bracket{a+b}=\bracket{a}+\bracket{b}`.



```
␣ We define  $[a]$  to be the equivalence class of  $a$ .  
This is a congruence:  $[a + b] = [a] + [b]$ .
```

Détails

- il ne faut pas utiliser la syntaxe "... " en mode math, car (1) elle ne s'imbrique pas bien et (2) elle est incompatible avec certains package mathématiques qui utilisent le symbole " comme tikzcd.
- `\intro*` ne marche pas dans l'environnement `align*`; on peut faire un `\phantomintro` à la place avant.
- il arrive que `knowledge` affecte la typographie (indices, exposants).

Portées multiples

```
\newcommand\var[1]{\kl[{\var{#1}}]{#1}}
```

```
\begin{theorem}\AP\label{thm:fermat3}
  \knowledge{\var{x}}{notion}
  \knowledge{\var{y}}{notion}
  \knowledge{\var{z}}{notion}
  Let $\intro*\var{x}$, $\intro*\var{y}$
  and $\intro*\var{z}$, be positive integers, then
  
$$\l[\var{x}^3 \neq \var{y}^3 + \var{z}^3 \ . \ ]$$

\end{theorem}
```

Here \var{x} is undefined.

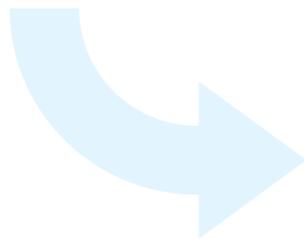
```
\knowledgeimport{thm:format3}
```

But here, \var{x} points to `\Cref{thm:format3}`.

Nécessite de charger `knowledge` avec l'option « `scope` ».

`\label` permet de définir des portées de même nom.

`\knowledgeimport` permet d'ajouter une portée pour qu'elle soit automatiquement cherchée.



Théorème 1. *Let x , y and z , be positive integers, then*

$$x^3 \neq y^3 + z^3 .$$

Here x is undefined.

But here, x points to Theorem 1.

Conclusion

Knowledge permet de créer et de maintenir des liens entre des notions dans un document, ou de des liens vers l'extérieur.

Il permet avec assez peu d'effort de créer des document où toutes les utilisations de concepts sont liées au lieu de leur introduction.

Knowledge-clustering est une aide extrêmement précieuse pour accélérer le travail.

Sa conception est prévue pour ralentir le moins possible le travail de l'auteur.

Il reste encore beaucoup de place à l'amélioration...