Rapport de stage

Yang YANG

2021-06-10

Contents

Chapter 1

Introduction

Dans le cadre du dernier semestre de DUT Informatique, un stage de 10 semaines est demandé pour la validation du diplôme. CEIPI m'a donné l'opportunité de réaliser mon stage à Strasbourg. Monsieur Julien Gossa exerce la fonction de responsable pédagogique et Monsieur Franck Macrez est mon tuteur professionnel.

Avant le stage, grâce aux explications de M. Gossa et un exemple préliminaire placé sur github, j'avais une compréhension globale pour ce stage. Le sujet est évalué le rythme des réformes par l'exploitation des dépôts git des codes. Et les résultats vont être visualisé pour les professionnels du droit.

Et le premier jour du stage, j'ai visité le bâtiment du CEIPI avec M. Gossa et rencontré M. Macrez. Après deux heures de communication, j'ai compris mieux les attentes du CEIPI, cela nous aide à l'élaboration de plans et le choix des outils.

Après ce rencontre, M. Gossa a créé un calendrier et publié des tâches spécifiques sur github de temps en temps pour contrôler la progression du stage. En effet, à cause de la situation sanitaire je fais mon travail à distance et le discord est le media de communication, mais il y avait aussi des réunions physiques lorsque nécessaire.

Les 350 heures de travail, qui représentes donc les 35 heures hebdomadaires d'un stage de dix semaines ont été effectué sur deux mois.

Ce rapport présentera la manière dont a été traité le sujet, les outils utilisés, les résultats que l'on a obtenus mais aussi comment surmonter certains problèmes techniques. En effet, pendant tout le processus de stage.

Chapter 2

Remerciement

Je remercie toutes les personnes qui m'ont soutenu personnellement ainsi que professionnellement avant et au cours de ce stage.

Tout d'abord, je remercie mon tuteur à l'IUT, **M. Gossa**, qui était toujours enthousiasme et patient pour répondre mes questions et proposer des solutions pratiques et qui a pris beaucoup de temps pour me guider et m'aider à organiser mon stage.

Je remercie également mon tuteur professionnel,**M. Macrez**, qui m'a emmené visiter et a présenté le CEIPI, et a expliqué les attentes du CEIPI pour ce stage.

J'adresse mon remerciement à **Mme. Kloess** à l'IUT et **Mme. Laplanche** au CEIPI qui s'ont occupé tous mes documents administratifs.

Finalement, j'aimerais remercier le jury qui écoute ma soutenance et qui me laisse une chance d'obtenir mon diplôme.

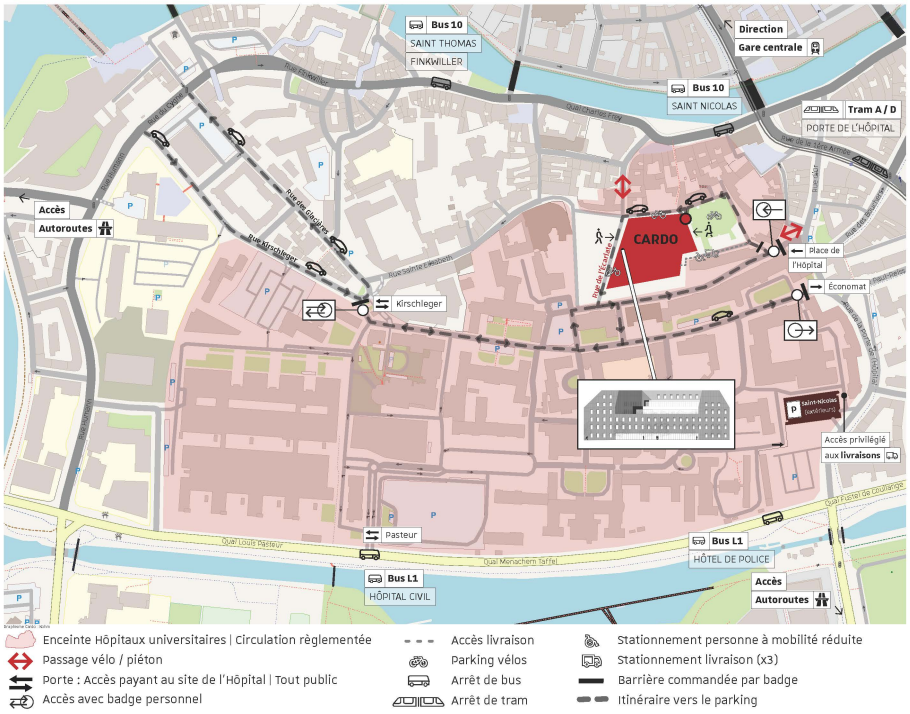
Chapter 3

Présentation du CEIPI

3.1 Informations Générales

CEIPI (Centre d'études internationales de la propriété intellectuelle) est une composante sous forme d'institut de l'université de Strasbourg, créer en 1963, à l'initiative des professeurs Daniel Bastien et Hubert Forestier. Dès sa création, le CEIPI s'est donnée la mission de former des spécialistes du droit de la propriété intellectuelle qui seront chargés d'exercer les différentes professions dans le domaine de la propriété intellectuelle.

CEIPI s'est installé le 2 mars 2020 dans un nouveau bâtiment situé dans l'enceinte de l'Hôpital civil à Strasbourg



PLAN D'ACCÈS

Figure 3.1: Location du CEIPI

(#fig:Location du CEIPI)

3.2 La structure

Le centre était composé de trois sections : La section française dispense aux spécialistes français un enseignement en matière de propriété intellectuelle à l'échelle nationale et internationale.

La section internationale consacre son programme de formation au contrat de licence et s'adressait aux spécialistes français et étrangers, désirant acquérir des connaissances nécessaires de droit international.

Le laboratoire de Recherche du CEIPI, créé en 2006, sa dénomination officielle est UR 4375-Laboratoire de recherche du CEIPI. Il coordonne des activités variées pour la mission de réflexion quant à l'évolution du droit de la propriété intellectuelle dans la société de la connaissance.

Plus des informations sur le site du CEIPI :https://www.ceipi.edu/

Chapter 4

Le sujet du stage

Le sujet de mon stage est quantification les évolutions législatives, en particulière pour les évolutions de la propriété intellectuelle. Les données pertinentes seront utilisées dans les recherches du CEIPI.

4.1 D'où viennent les données

Il existe trois dépôts git présentent les données législatives françaises : **Legifrance** : https://github.com/legifrance

Etalab : https://github.com/etalab/codes-juridiques-francais

Archeo Lex: https://archeo-lex.fr/

Nous avons finalement choisi Archeo Lex. Bien que ses données ne soient pas aussi précises qu'EtabLab, mais il régulièrement mis à jour les données. En plus, ses données sont organisées bien, chaque code est placé dans un fichier séparé et nous pouvons trouver toutes les versions dans l'ordre.

Figure 4.1: Les données dans Archeo Lex

4.2 Traitement du sujet

Après discussion, le stage a été divisé en deux parties. L'un est l'exploration de données, dont le but est de créer une application qui peuvent générer des documents csv différentes en modifiant la ligne de commande.

L'autre est la visualisation des données.

4.3 Choix des outils

4.3.1 Langage

1.**Python** : nous avons choisi python comme langage d'exploration de données, car il contient de nombreux packages pouvant être appelés et il n'est pas difficile à apprendre. Parmi eux, le package le plus important que nous utilisons est git-python*, qui peut être utilisé pour

2.**R et Rmd**: Nous avons choisi R comme langage de dessin. tidyverse, ggraphe et igraphe sont les plus fréquemment utilisés parmi les nombreux packages du langage R. La partie de code de R est stockée sous la forme d'un fichier Rmd, et le rapport que vous en train de lire et la diapositive de la soutenance sont tous générés par le fichier Rmd.

4.3.2 IDE

1.**Vscode** : l'outil d'environnement de développement le plus populaire. Il est très flexible et il est aussi techniquement abouti et très stable, je l'ai utilisé pour développer l'application python.

2.**Rstudio**: un environnement de développement gratuit, libre et multiplateforme pour R.il sert au traitement de données et à l'analyse statistique

Chapter 5

Application archeolex_excavation.py

5.1 Présentation Générale

Rôle : archeolex_excavation.py est utilisée pour fouiller des dépôts git Archéo Lex et générer des fichiers csv. Nous pouvons générer différents fichiers csv en modifiant les paramètres dans la ligne de commande.

Usage : archeolex_excavation.py [-h] [-d YYYY-MM-DD] [-f fichier.csv] [-t] [-v] diff|check| |stats [code ...]

Example:

```
archeolex_excavation.py stats code_civil -d last -t -s1 -f test.csv
```

5.1.1 Les paramètres

5.1.1.1 Positional arguments

nom de paramètre	signification
diff/check/status	**Le traitement à effectuer**
diff	obtenir les informations de modification par article
check	détecter des erreurs des codes
status	Les informations de sous-section d'une section, ainsi que le nombre de lignes et de mots dans cette section.Les paramètres d'entrée -s1 à -s6 (voir ci-dessous) pour confirmer le niveau de cette section.
codes	**La liste des codes à fouiller**

5.1.1.1.1 Les niveaux généraux de la structure des codes Partie

Sous_partie

Livre

Titre

Chapitre

5.1.1.2 Optional arguments

nom de paramètre	signification
-h, –help	montrer le message de help et quitter
-d YYYY-MM-DD, –datelimit YYYY-MM-DD	définir une date maximum pour la fouille

nom de paramètre	signification
-f nom.csv, –file nom.csv	écrit les données dans ce fichier csv (sortie standard par défaut)
-t, –fulltext	détecter les noms entiers des section
-s1	Obtenir les informations de chaque chapitre
-s2	Obtenir les informations de chaque livre
-s3	Obtenir les informations de chaque titre
-s4	Obtenir les informations de chaque sous partie
-s5	Obtenir les informations de chaque partie
-s6	Obtenir les informations de chaque code

5.1.1.3 Quelques fichies csv générés

code	version	date	partie	sous_partie	livre	titre	chapitre	article	type
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre Ier : Des cas de divorce	233	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre Ier : Des cas de divorce	238	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre Ier : Des cas de divorce	246	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre Ier : Des cas de divorce	247-2	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	251	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	252	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	253	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	254	Modification
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	257	Suppression
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	257-1	Suppression
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	257-2	Suppression
code_civil	2b5d875	01/01/2021	Législative	Na	Livre Ier : Des personnes	Titre VI : Du divorce	Chapitre II : De la procédure du divorce judiciaire	258	Suppression

Figure 5.1: les informations de modification par article

5.1.1.3.1 Example 1:

5.1.1.3.2 Example 2: L'application permet de détecter des erreurs de deux types :

- doublon : articles apparaissant deux fois dans un code ;
- inversion : deux articles consécutifs dont la numérotation n'est pas croissante. Cette détection d'erreur est imparfaite, et n'exclu ni faux-positifs ni faux-négatifs. La date correspond à la version la plus ancienne à laquelle l'erreur a été détectée.

Les erreurs détectées sur un échantillon de codes se trouvent dans le fichier errors.csv, au format suivant :

code	version	date	partie	sous_partie	livre	titre	chapitre	article	type
code_civil	00f14be	1803-04-29	Législative	NA	8	1	9	819	inversion 842
code_civil	d48a9bd	1803-05-13	Législative	NA	9	0	5	905	inversion 1095
code_civil	f7a5147	1804-02-17	Législative	1	3	1	6	1316	inversion 1369
code_civil	74471bb	1804-02-24	Législative	2	0	2	4	2024	inversion 2027
code_civil	d86cb5f	1804-03-17	Législative	1	4	9	2	1492	inversion 1523
code_civil	63bf723	1804-03-26	Législative	2	0	6	2	2062	inversion 2070
code_civil	8edc69c	1804-03-29	Législative	NA	6	6	4	664	inversion 665
code_civil	a22e7b9	18/02/1938	Législative	NA	9	1	0	910	inversion 1095
code_civil	981fe53	07/01/1955	Législative	2	1	0	7	2107	inversion 2113
code_civil	ebc294e	08/01/1959	Législative	NA	9	3	7	937	inversion 942
code_civil	684179a	15/06/1965	Législative	NA	4	0	2	402	inversion 448
code_civil	eb95e99	01/02/1966	Législative	NA	8	1	8	818	inversion 820

Figure 5.2: détecter des erreurs des codes

5.1.1.3.3 Example 3

5.1.1.3.4 Example 4

code	date	partie	sous_partie	livre	nb_titres	nb_chapitres	nb_articles	nb_alineas	nb_mots
code_civil	01/01/2021				1	1	7	12	286
code_civil	01/01/2021			Livre Ier : Des personnes	15	52	808	1859	69235
code_civil	01/01/2021			Livre II : Des biens et des différentes modifications de la propriété	4	8	194	316	10170
code_civil	01/01/2021			Livre III : Des différentes manières dont on acquiert la propriété	21	68	1592	2804	90122
code_civil	01/01/2021			Livre IV : Des sûretés	2	12	229	518	17919
code_civil	01/01/2021			Livre V : Dispositions applicables à Mayotte	4	1	39	112	3135
code_civil	16/12/2020				1	1	7	12	286
code_civil	16/12/2020			Livre Ier : Des personnes	15	52	812	1862	69291
code_civil	16/12/2020			Livre II : Des biens et des différentes modifications de la propriété	4	8	194	316	10170
code_civil	16/12/2020			Livre III : Des différentes manières dont on acquiert la propriété	21	68	1592	2804	90122
code_civil	16/12/2020			Livre IV : Des sûretés	2	12	229	518	18094
code_civil	16/12/2020			Livre V : Dispositions applicables à Mayotte	4	1	39	112	3135

Figure 5.3: Les informations de sous-section des livres, ainsi que le nombre de lignes et de mots pour chaque livres.

code	date	nb_partie	nb_sous_pa	nb_livre	nb_titres	nb_chapitres	nb_articles	nb_alineas	nb_mots
code_civil	01/01/2021	1	1	5	47	142	2869	5621	190867
code_civil	16/12/2020	1	1	5	47	142	2873	5624	191098
code_civil	09/12/2020	1	1	5	47	142	2873	5624	191078
code_civil	01/09/2020	1	1	5	47	142	2873	5622	191056
code_civil	01/08/2020	1	1	5	47	142	2877	5630	191301
code_civil	14/02/2020	1	1	5	47	142	2877	5628	191117
code_civil	01/01/2020	1	1	5	47	142	2877	5626	191045
code_civil	30/12/2019	1	1	5	47	142	2877	5626	191195
code_civil	28/12/2019	1	1	5	47	142	2875	5620	190569
code_civil	15/12/2019	1	1	5	47	142	2875	5609	190212
code_civil	23/10/2019	1	1	5	47	142	2875	5609	190214
code_civil	21/07/2019	1	1	5	47	142	2876	5610	190271
code_civil	12/07/2019	1	1	5	47	142	2875	5607	189999

Figure 5.4: Les informations de sous-section des chapitres, ainsi que le nombre de lignes et de mots pour chaque chapitre.

5.2 Présentation technique

5.2.1 La structure des codes

L'idée de code python est **orientée objet**,il y a trois fichiers python

1.**Article.py**: un article doit être considéré comme un objet, et les attributs liés à l'article peuvent être trouvés dans cette classe. Par exemple, le chapitre, le livre , le code où il se trouve, sa date, etc. (voir l'UML dans la figure ci-dessous pour plus de détails). Les fonctions permettent de modifier ou d'obtenir la valeur des attributs d'article.

2.**ArcheoLexLog.py**: Le rôle de cette classe est de stocker des fonctions d'exploration et d'analyse de données. Cette classe appelle la classe Article et appelle également un package très important GitPython* pour obtenir des informations de git diff.

3.**archeolex_excavation.py**:Ce fichier appelle les deux classes Article et ArcheoLexLog, définit les paramètres et contient la fonction main.

5.2.2 problèmes rencontrés et solution

-1.Au début,l'idée était d'obtenir les informations de la structure de l'article à partir du nom. Par exemple : Article L312 -> L'article se trouve dans la partie législative, le troisième livre, le premier titre, le deuxième chapitre. Cependant, il existe huit lois et leurs noms d'articles ne sont pas liés à la structure : code_civil code_de_l'artisanat code_de_la_famille_et_de_l'aide_sociale code_de_procédure_civile code_de_procédure_pénale code_des_postes_et_des_communicet_pélectroniquedemaraire_demarale_demarique

Solution: parcourir le texte complet du diff pour obtenir les informations de structure correspondantes. Et le nom complet(fulltext) de la structure est directement écrit dans la table sans utiliser le nom numérique.

-2.Un changement structurel après un commit. Par exemple: une nouvelle sous_section est ajoutée

Solution: ajouter du code pour détecter et enregistrer ce changement.

-3.Les dépots git sur Archeo Lex ont des problème. Par exemple: il y a un problème avec l'ordre des articles, et le même article du même commit a été modifié plusieurs fois.

Solution: ajout d'un paramètre check pour détecter et afficher les erreurs, et envoyer les erreurs à Archeo Lex.

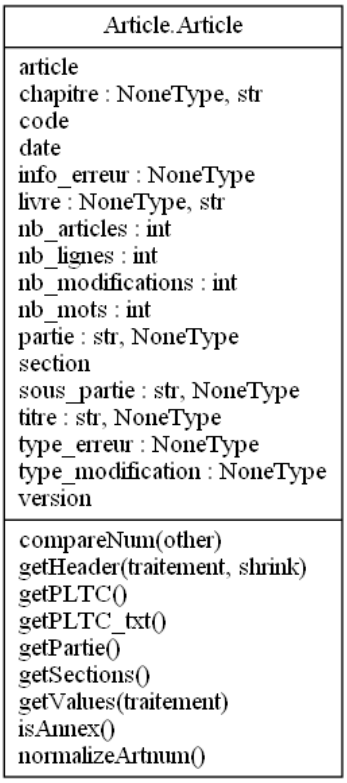


Figure 5.5: le diagramme UML pour Article.py

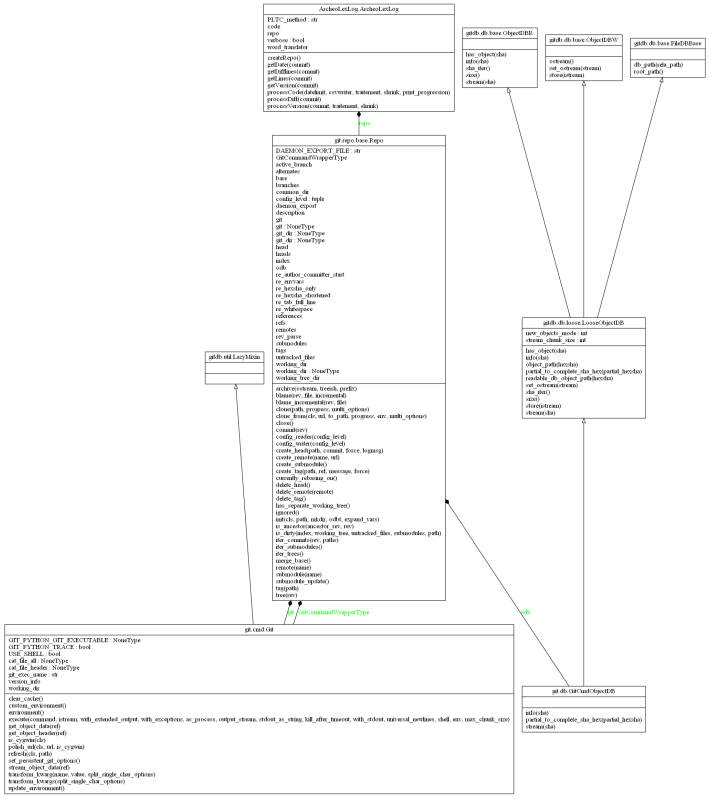


Figure 5.6: le diagramme UML pour ArcheoLexLog.py

Chapter 6

Les images générés

6.1 modification et des visualisations

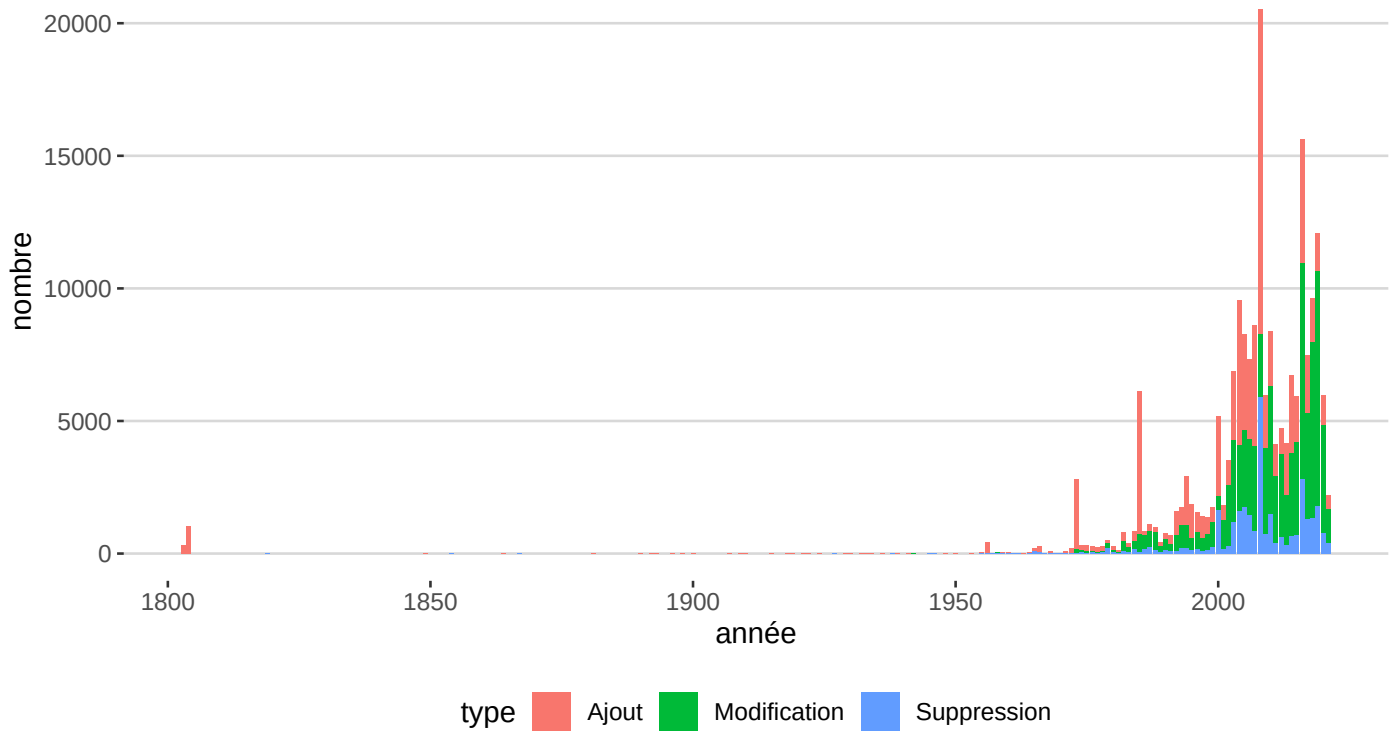
6.1.1 Liste des codes et modifications

code	nb_modifs	début	fin	parties	sous_parties	livres	titres	c
code_civil	6248	1803-03-15	2021-01-01	1	1	7	70	
code_de_commerce	16455	2000-12-14	2021-05-23	4	1	11	80	
code_de_l'action_sociale_et_des_familles	8513	2001-07-18	2021-05-21	3	1	8	49	
code_de_l'éducation	10754	2000-12-14	2021-05-24	2	6	15	97	
code_de_la_consommation	6235	1994-01-04	2021-04-16	4	1	16	63	
code_de_la_propriété_intellectuelle	3227	1993-01-01	2021-05-14	2	6	17	27	
code_de_la_recherche	441	2004-08-11	2021-01-01	1	1	5	21	
code_de_la_santé_publique	51280	1953-10-27	2021-05-27	6	10	87	290	
code_de_la_sécurité_intérieure	4700	2012-12-23	2021-05-27	2	1	9	59	
code_de_la_sécurité_sociale	40633	1961-01-12	2021-05-23	5	1	37	185	
code_du_travail	48381	1973-07-11	2021-05-27	6	9	76	342	
code_pénal	3235	1992-07-23	2021-05-27	2	1	11	31	

6.1.2 Pourcentage de différents types de modifications

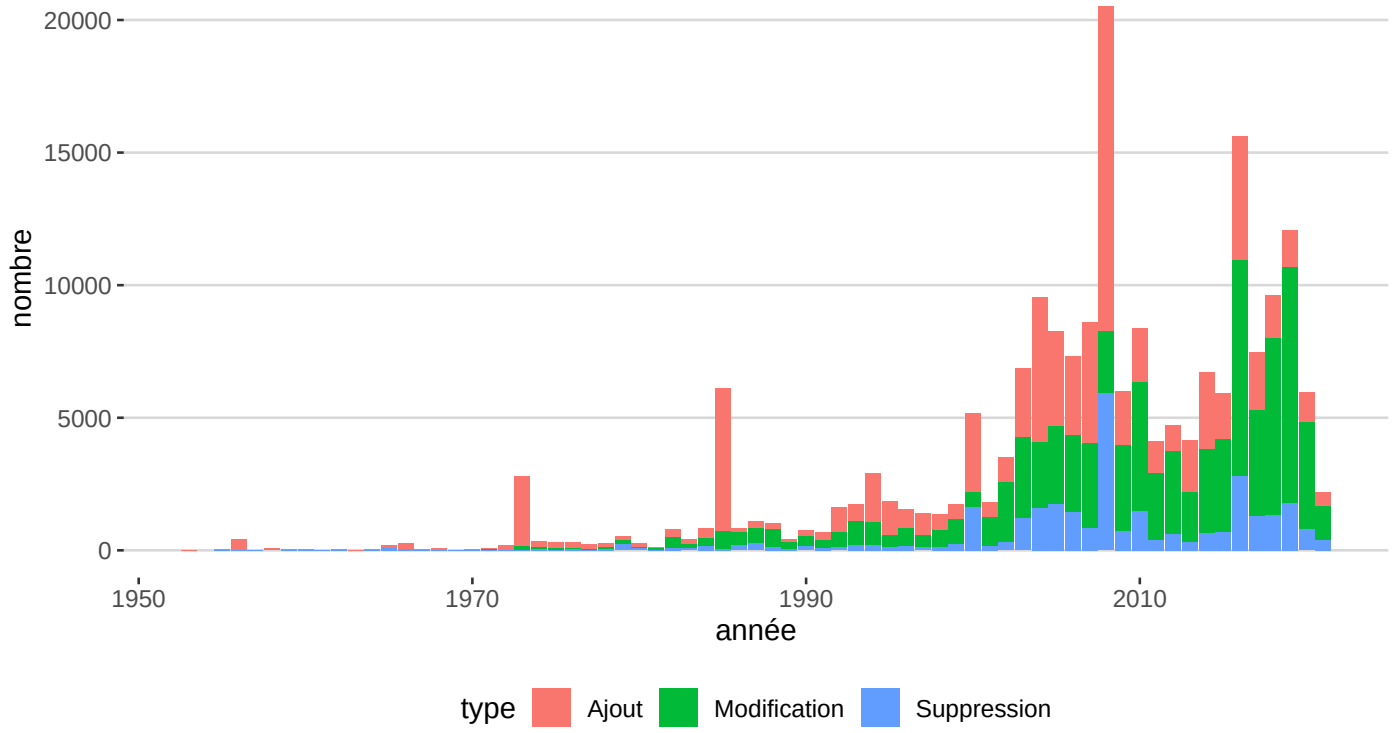
6.1.2.1 Toutes les années

```
## `summarise()` has grouped output by 'année'. You can override using the `.groups` argument.
```



6.1.2.2 Depuis 1950

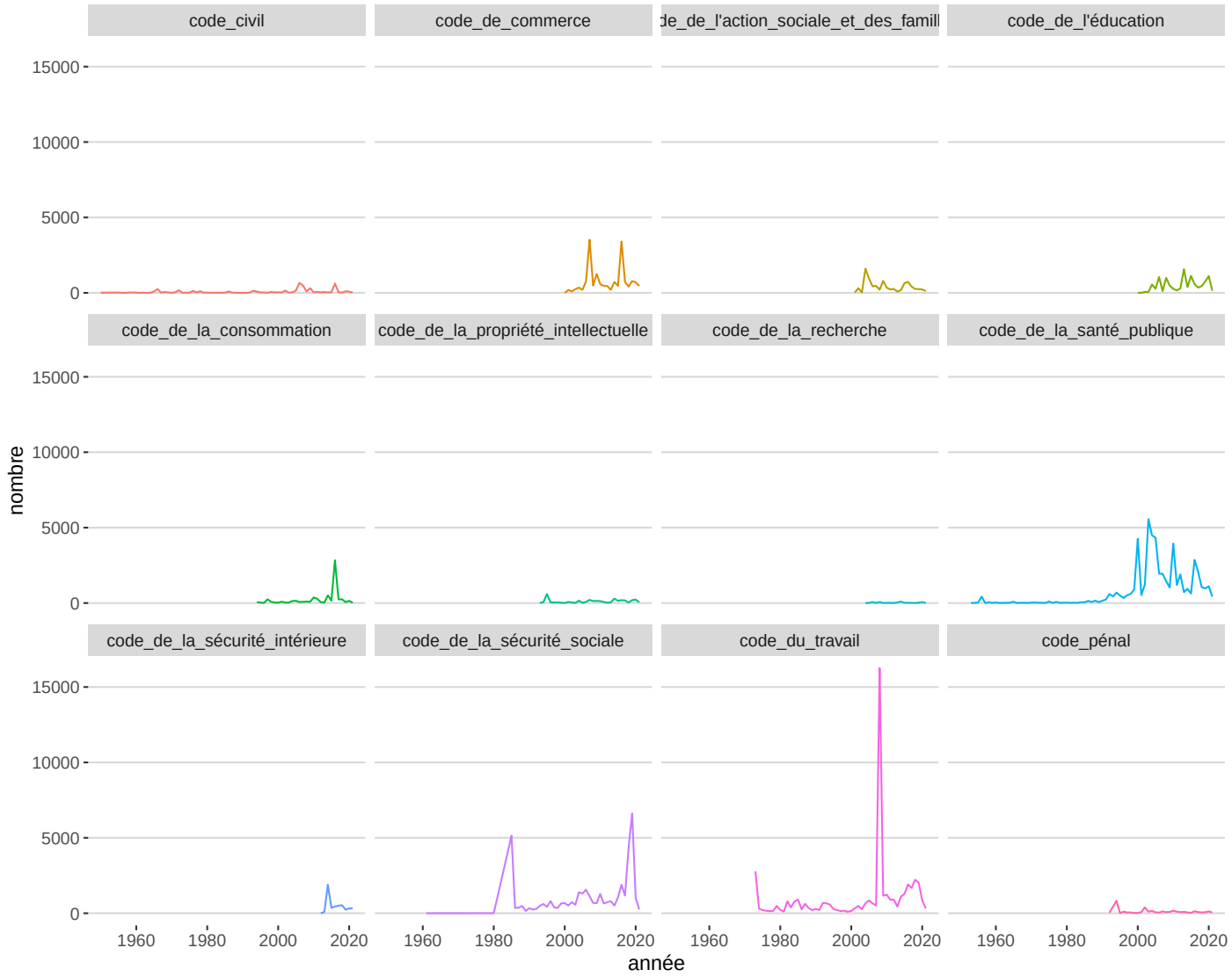
```
## `summarise()` has grouped output by 'année'. You can override using the `.groups` argument.
```



6.1.3 Nombre de modifications de chaque année

6.1.3.1 Depuis 1950

```
## `summarise()` has grouped output by 'année'. You can override using the `.groups` argument.
```



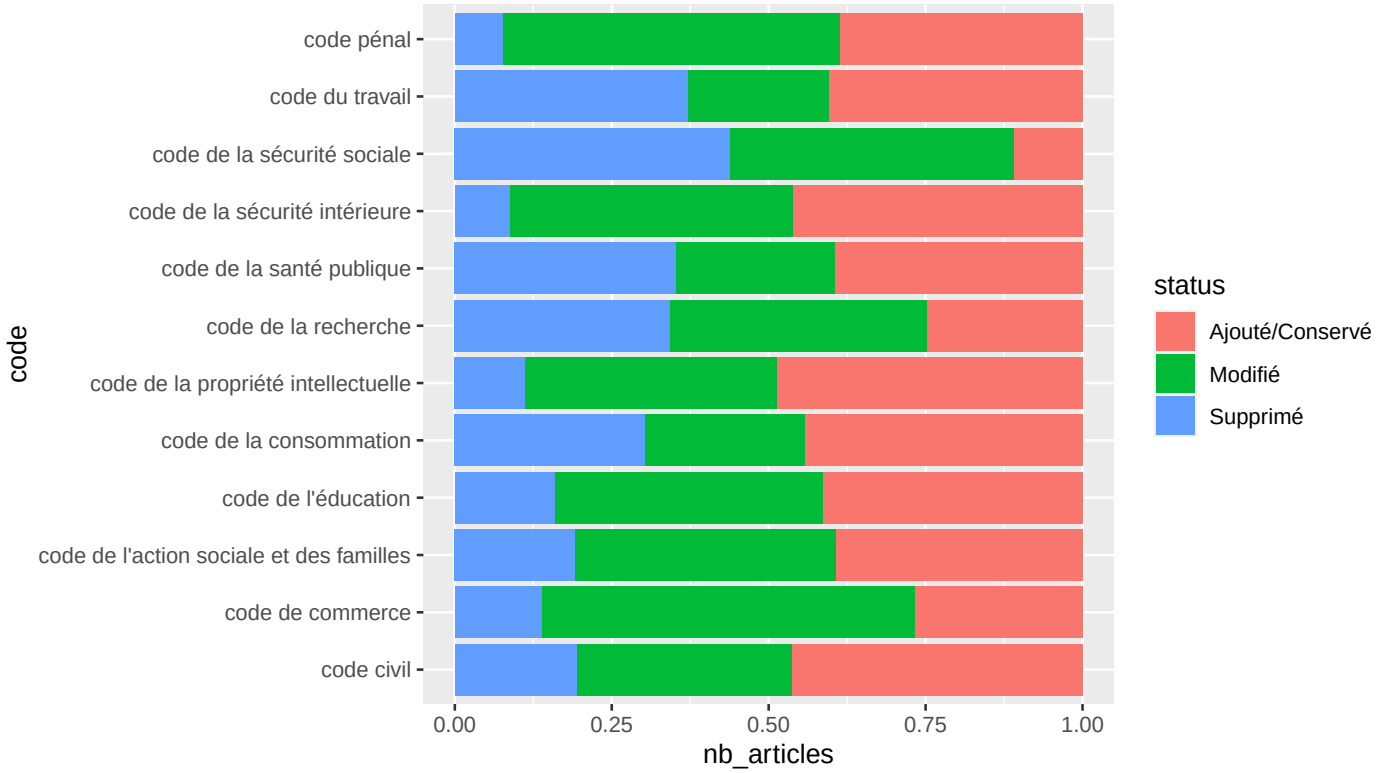
6.1.3.2 Depuis 1999

```
## `summarise()` has grouped output by 'année'. You can override using the `.groups` argument.
```

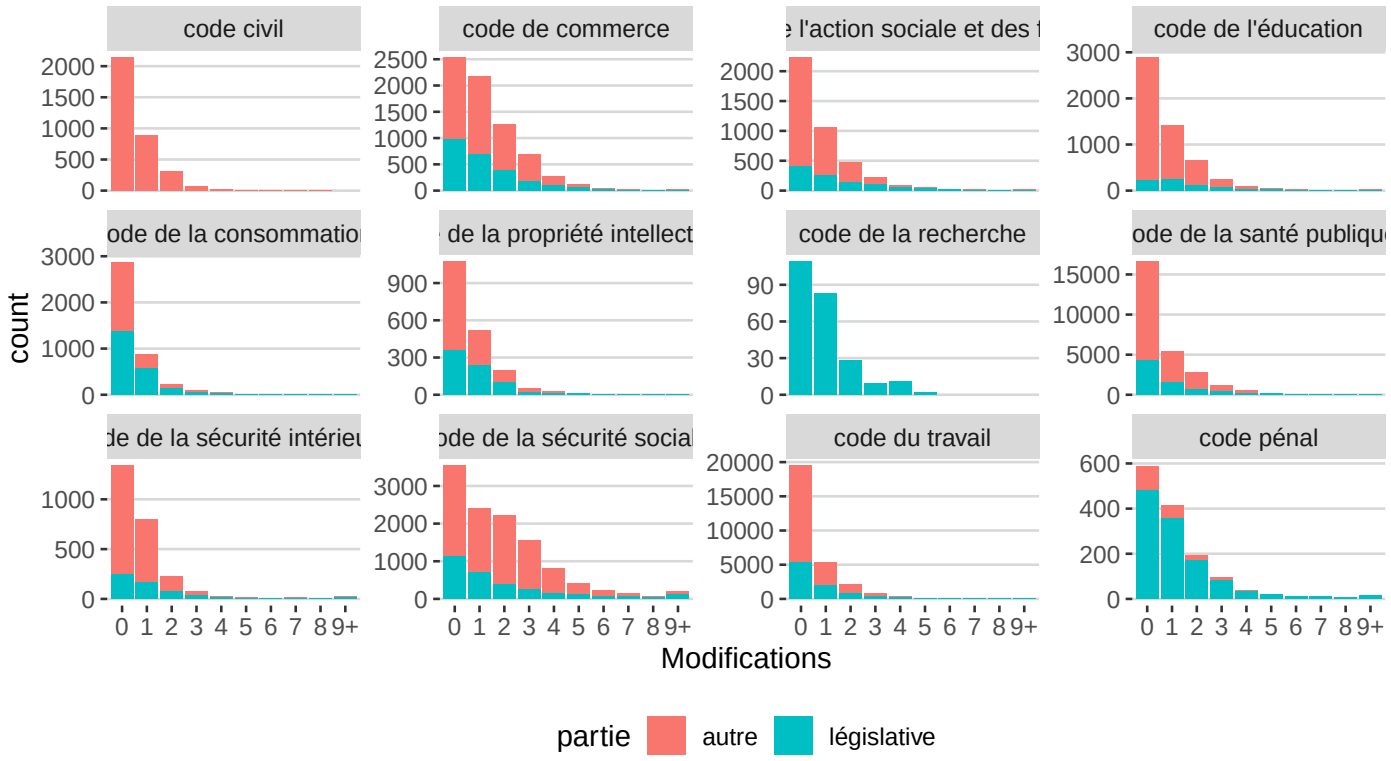

```
## `summarise()` has grouped output by 'code', 'partie', 'article'. You can override using the `.groups` argu
```

6.1.4 Etat actuel

```
## `summarise()` has grouped output by 'code'. You can override using the `.groups` argument.
```



6.1.5 Nombre modifications par articles

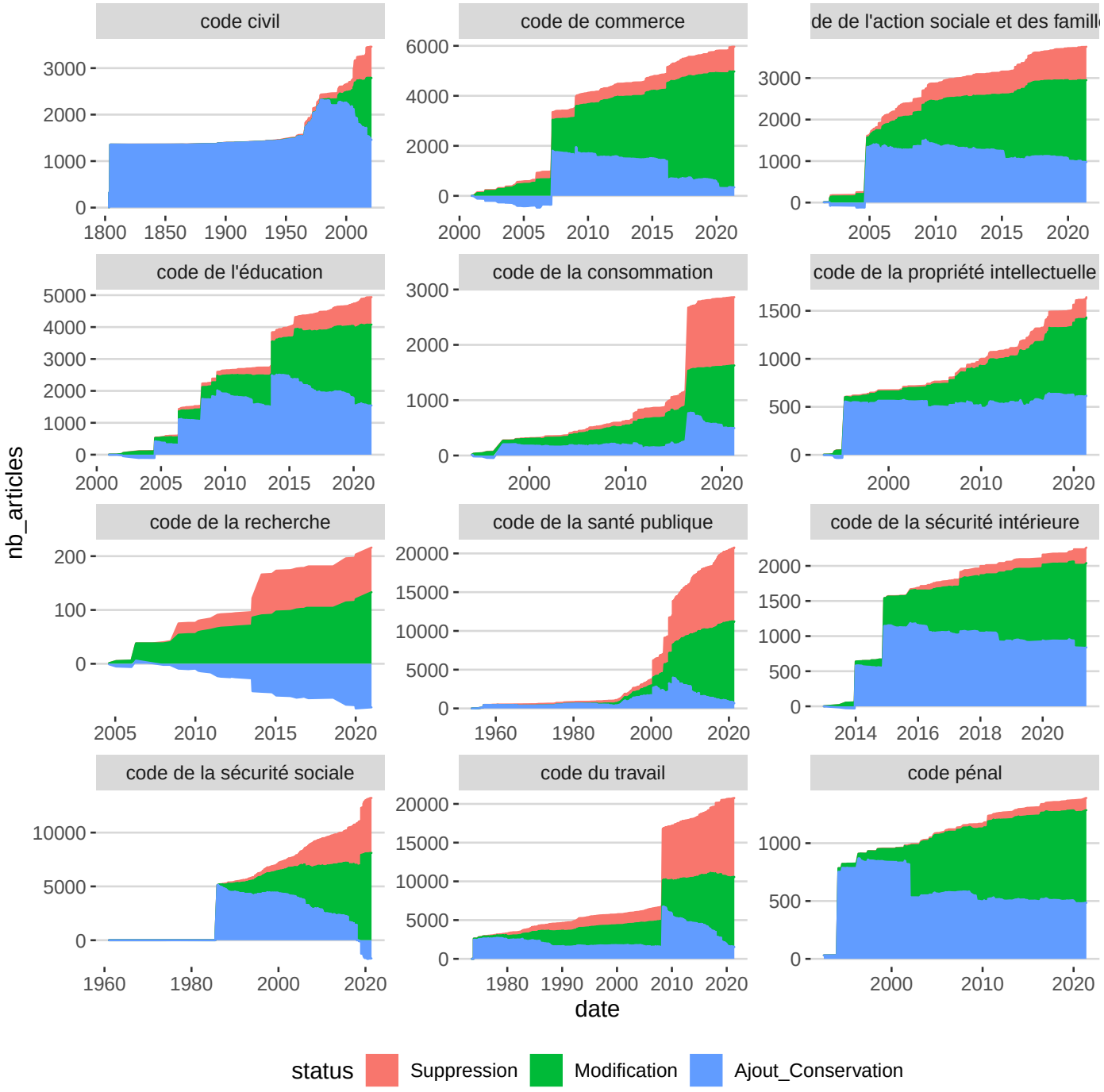



```
## Warning: Unknown levels in `f`: Pr xistence
```

```
## `summarise()` has grouped output by 'code', 'article'. You can override using the `.groups` argument.
```

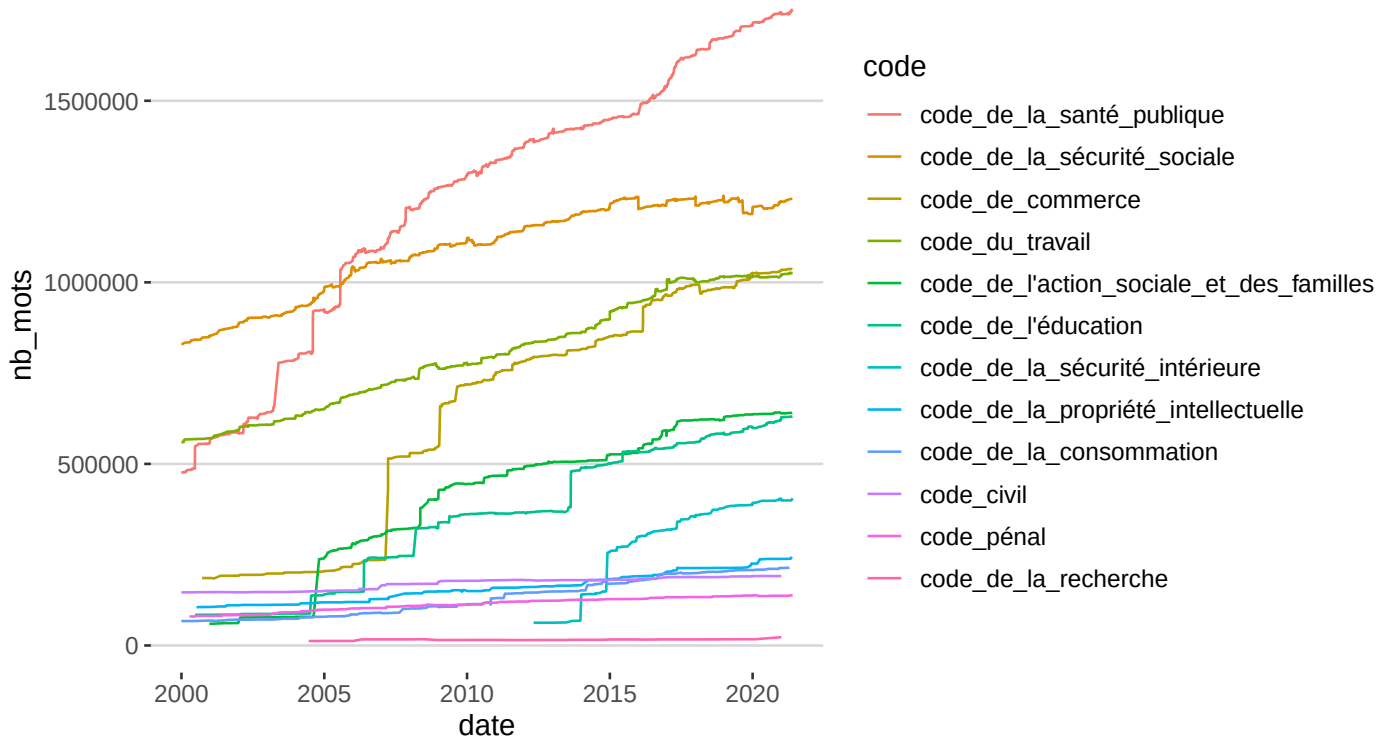
```
## `summarise()` has grouped output by 'code', 'date'. You can override using the `.groups` argument.
```

6.1.6 Evolution de nombre de modifications

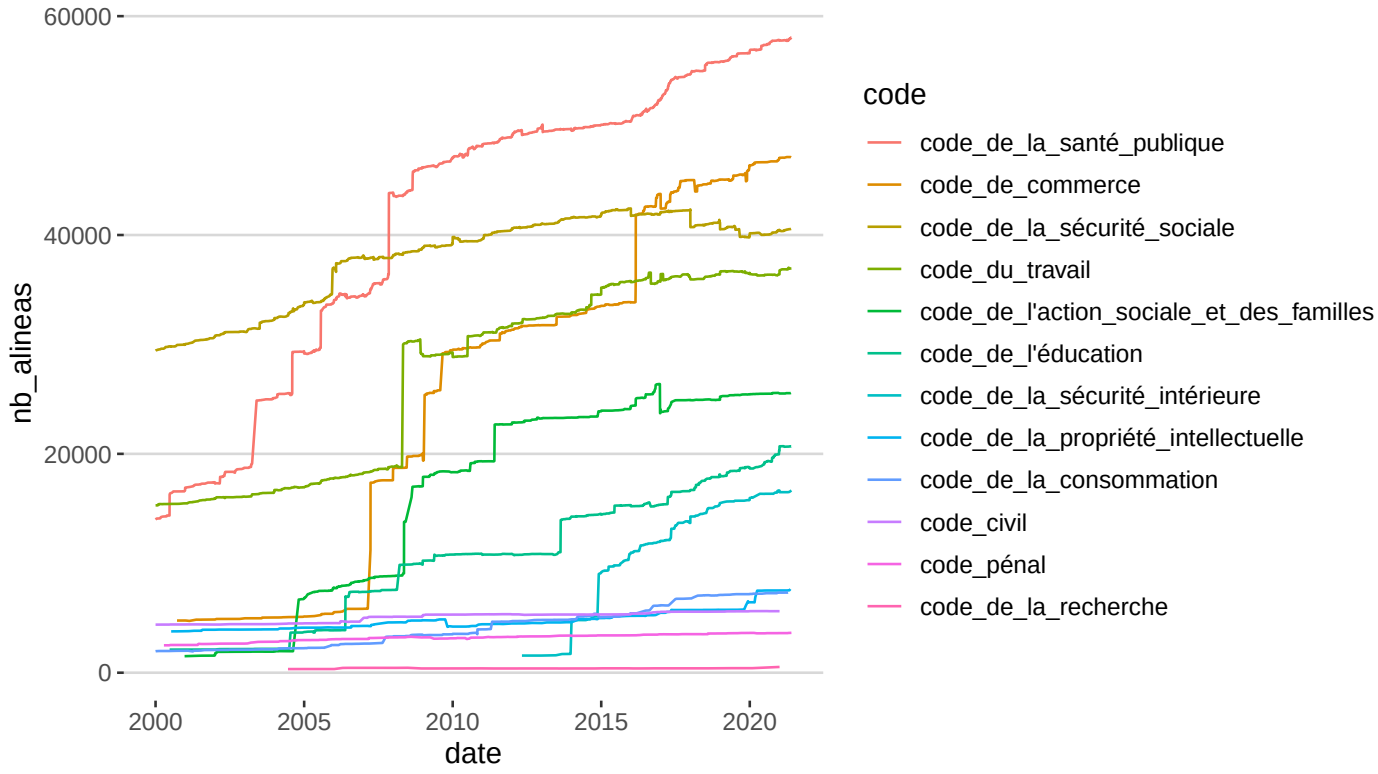


6.2 Evolution des volumes de codes

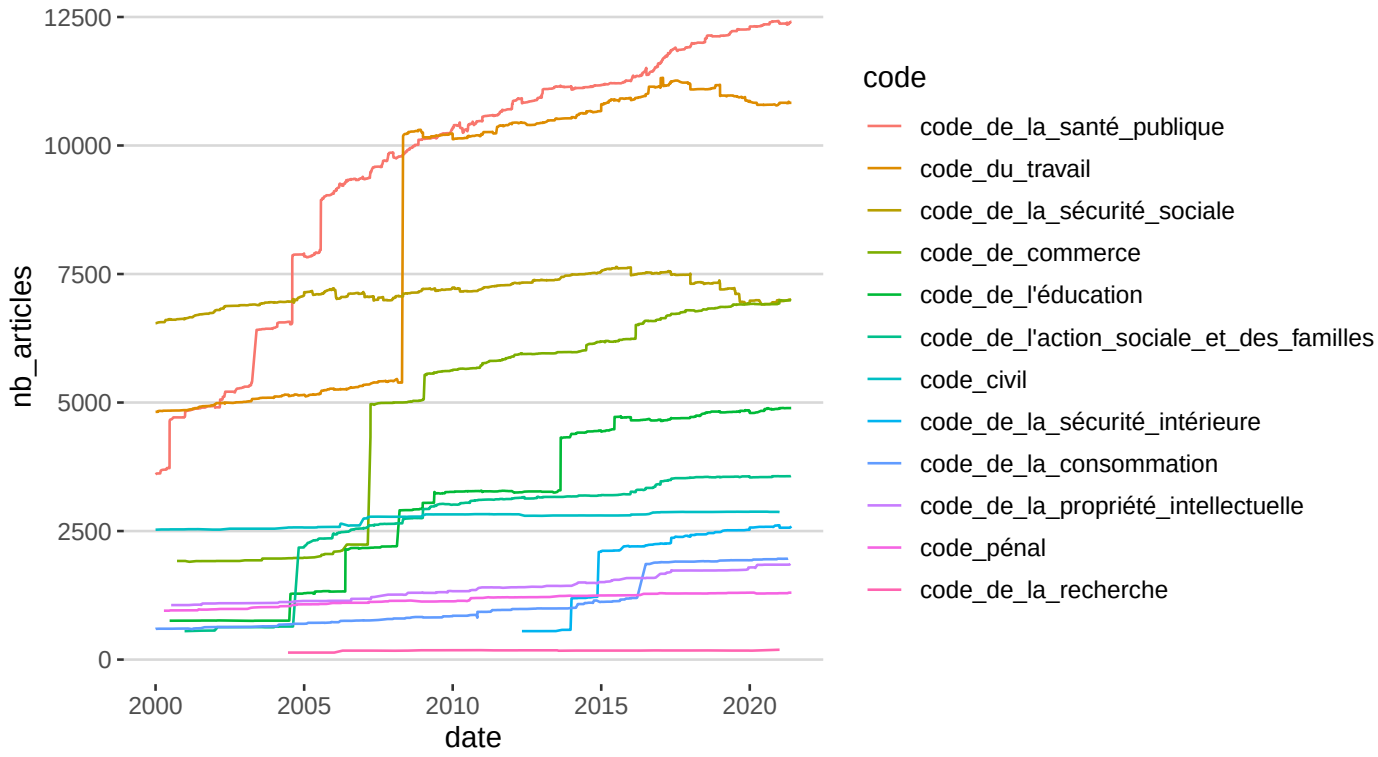
6.2.1 Nombre de mots



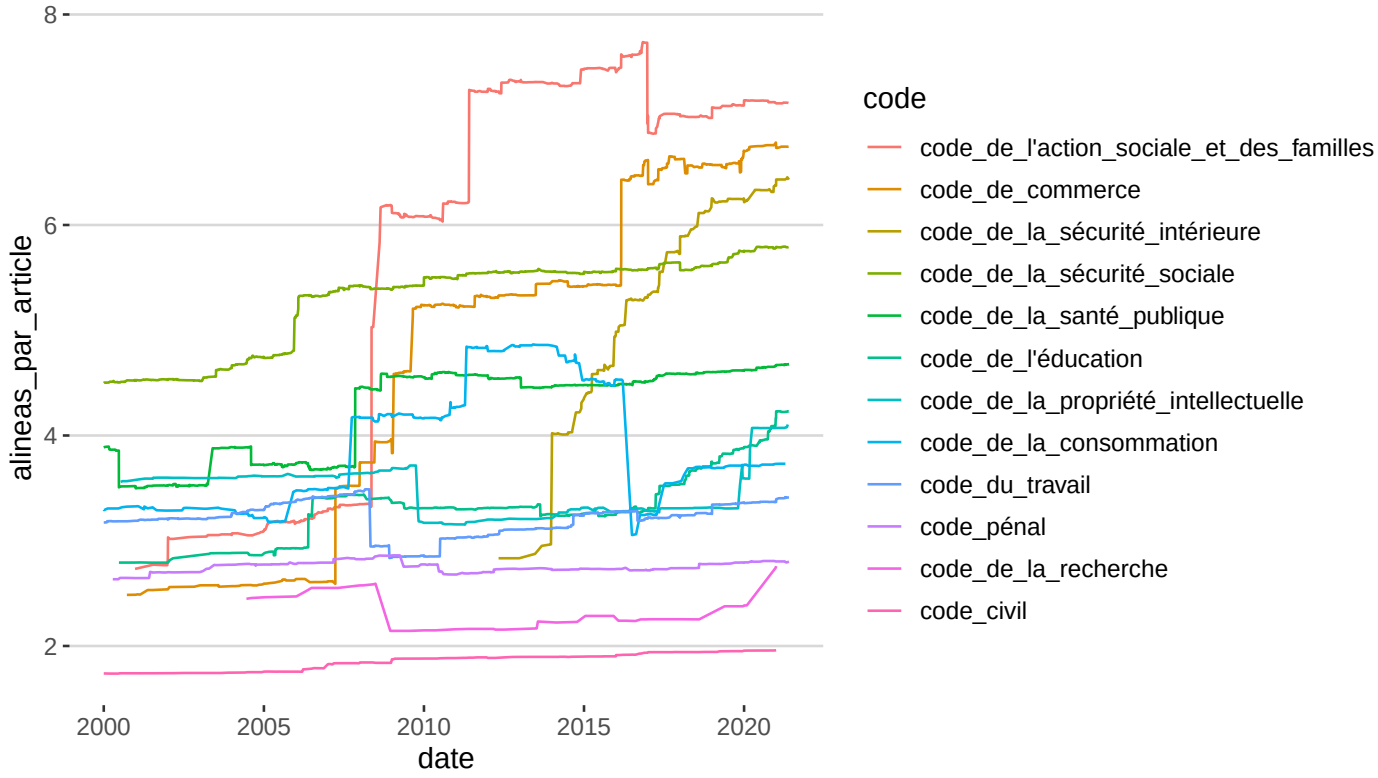
6.2.2 Nombre de lignes



6.2.3 Nombre d'article



6.2.4 Nombre moyen de lignes d'un article



6.2.5 Nombre moyen de mots d'un article

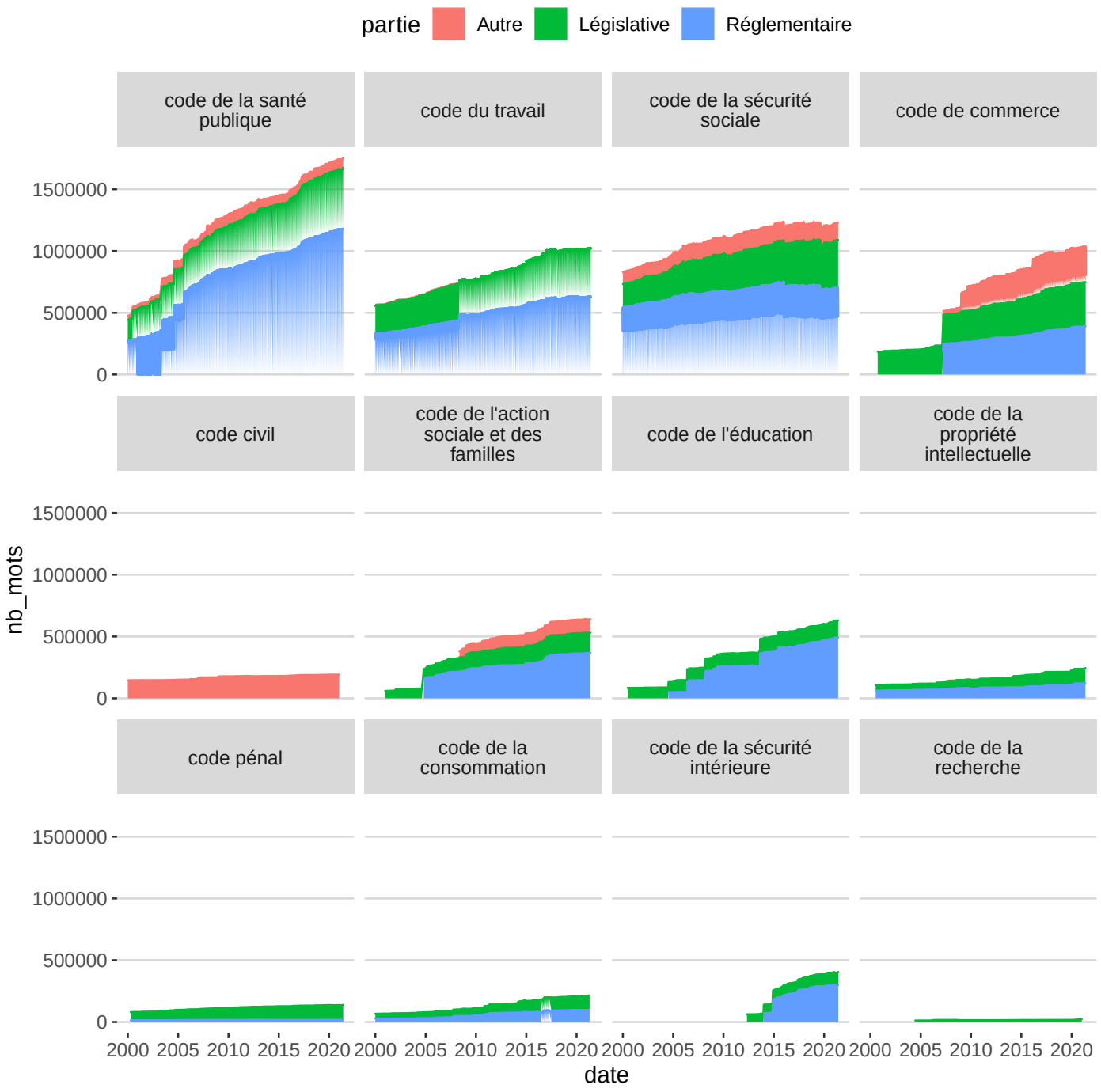
6.2.6 Nombre moyen de mots d'une ligne

6.3 Structure

6.3.1 Législative vs. Réglementaire

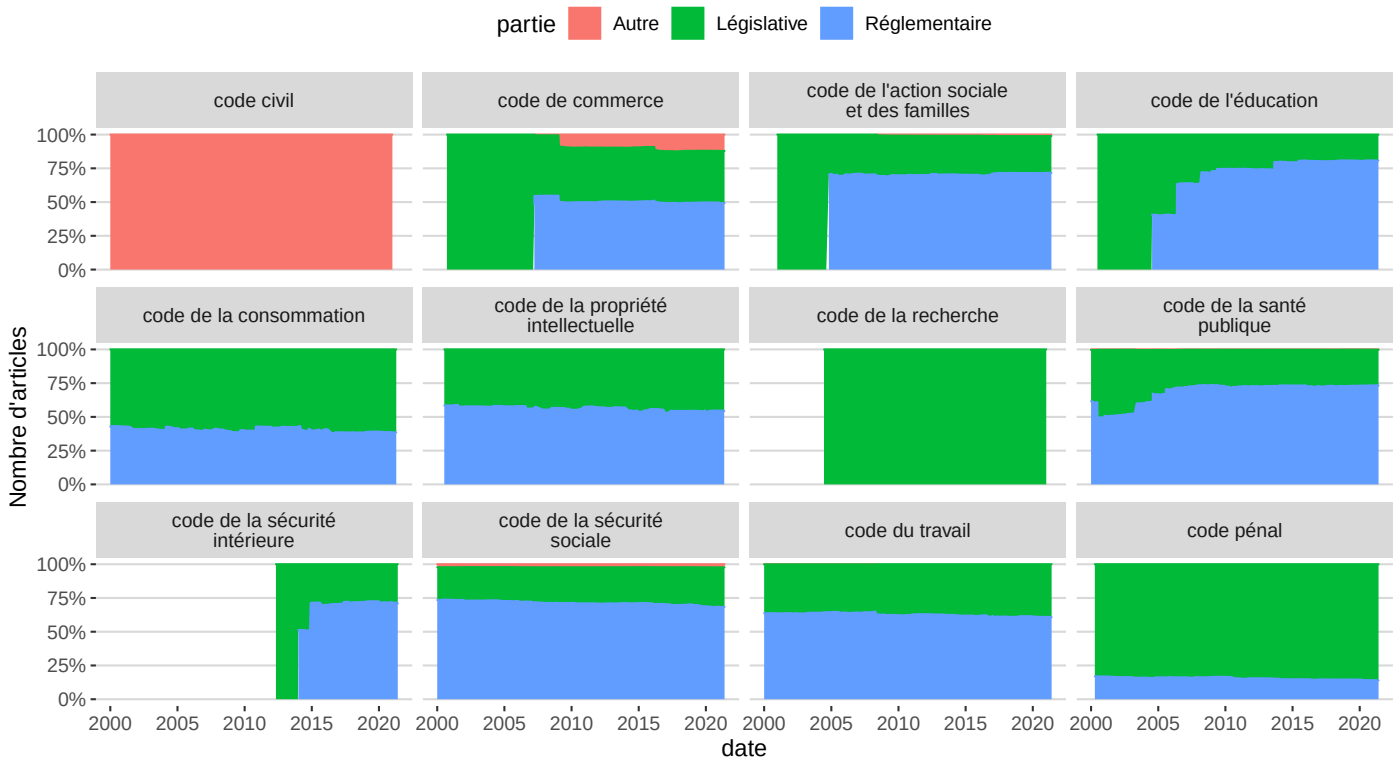
6.3.1.1 Nombre de mots de chaque partie(séparées)

6.3.1.2 Nombre de mots de chaque partie(combinées)



6.3.1.3 Pourcentage de chaque partie

```
## `summarise()` has grouped output by 'date', 'code'. You can override using the `.groups` argument.
```



6.3.2 Arborescence

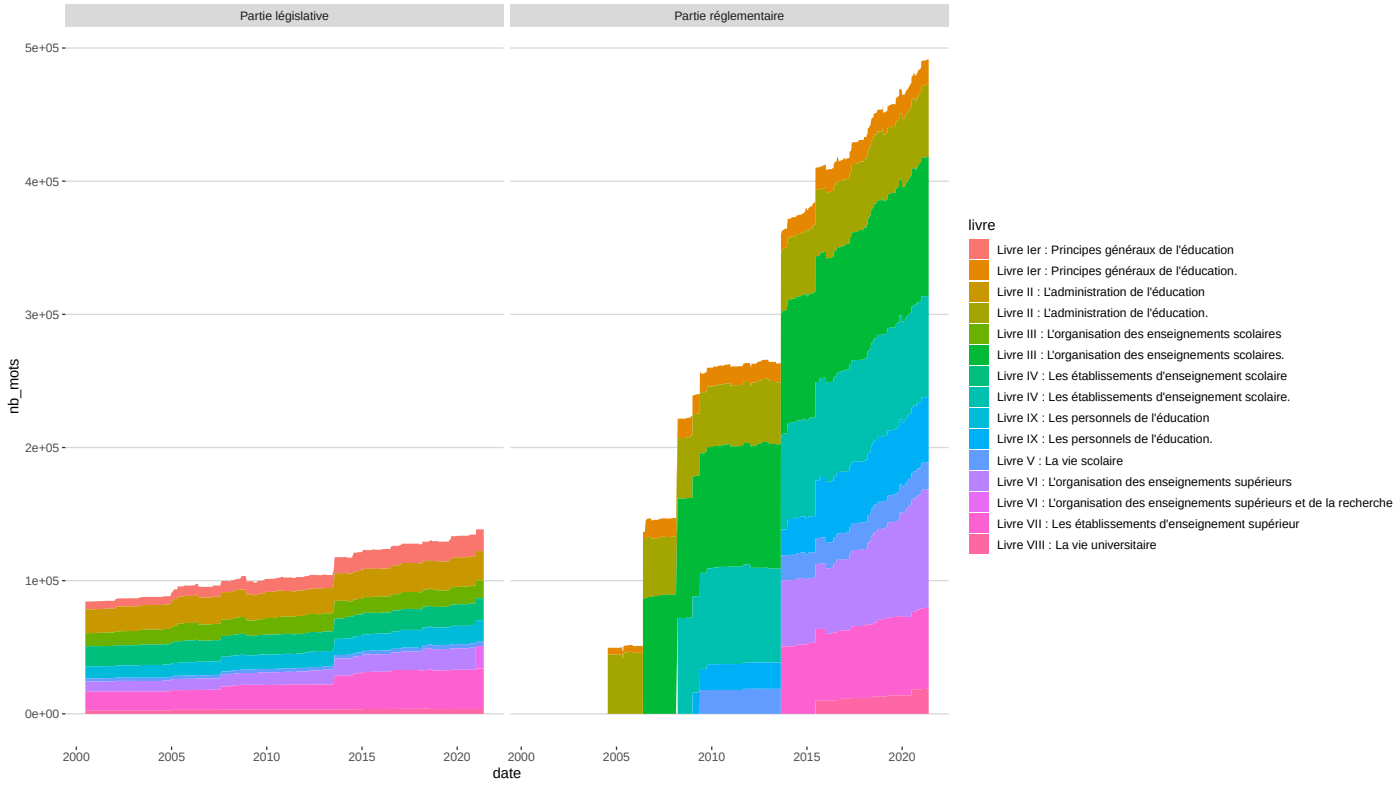
{r tree.edu, fig.width=15, fig.height=15, out.width=500, out.height=500, fig.align="center"} legiplot_tree("code de l'éducation")

{r tree.pro, fig.width=15, fig.height=15, out.width=500, out.height=500, fig.align="center"} legiplot_tree("Code de la propriété")

6.4 Modification pour chaque livre

6.4.1 Code de l'éducation

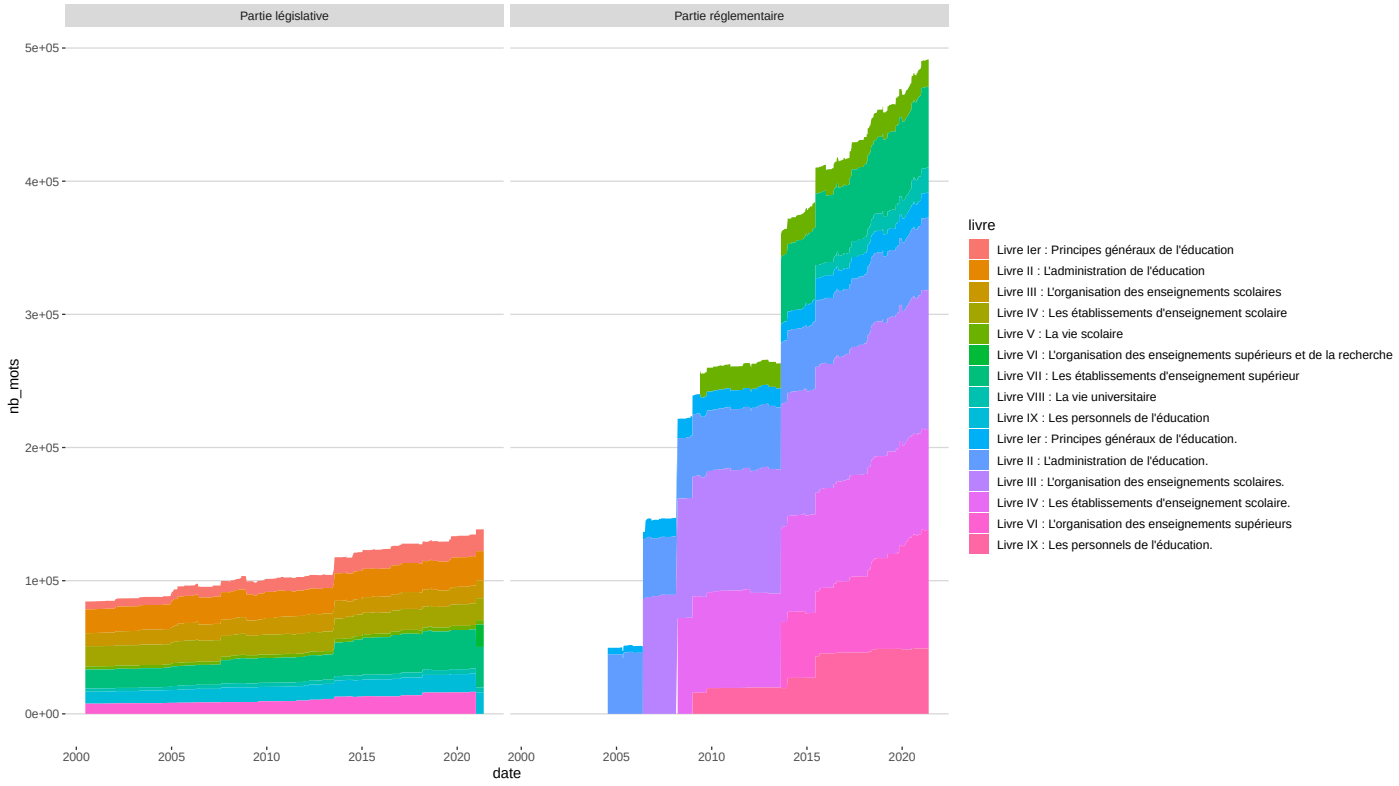
```
stats.det <- read.csv("../stats_shortlist_ts3.csv",encoding = 'UTF-8') %>%
  filter(code == "code_de_l'éducation") %>%
  mutate(date = as.Date(date))
stats.det %>%
  ggplot(aes(x=date,y=nb_mots,fill=livre)) +
  geom_area(aes(group=livre)) +
  facet_grid(.~partie)  +
  theme_hc() +
  theme(legend.position = "right")
```



Premier problème : les livres ne sont pas dans l'ordre : Livre IX < Livre V

Pour traiter ce problème, on force l'ordre des livres avec l'ordre dans le csv:

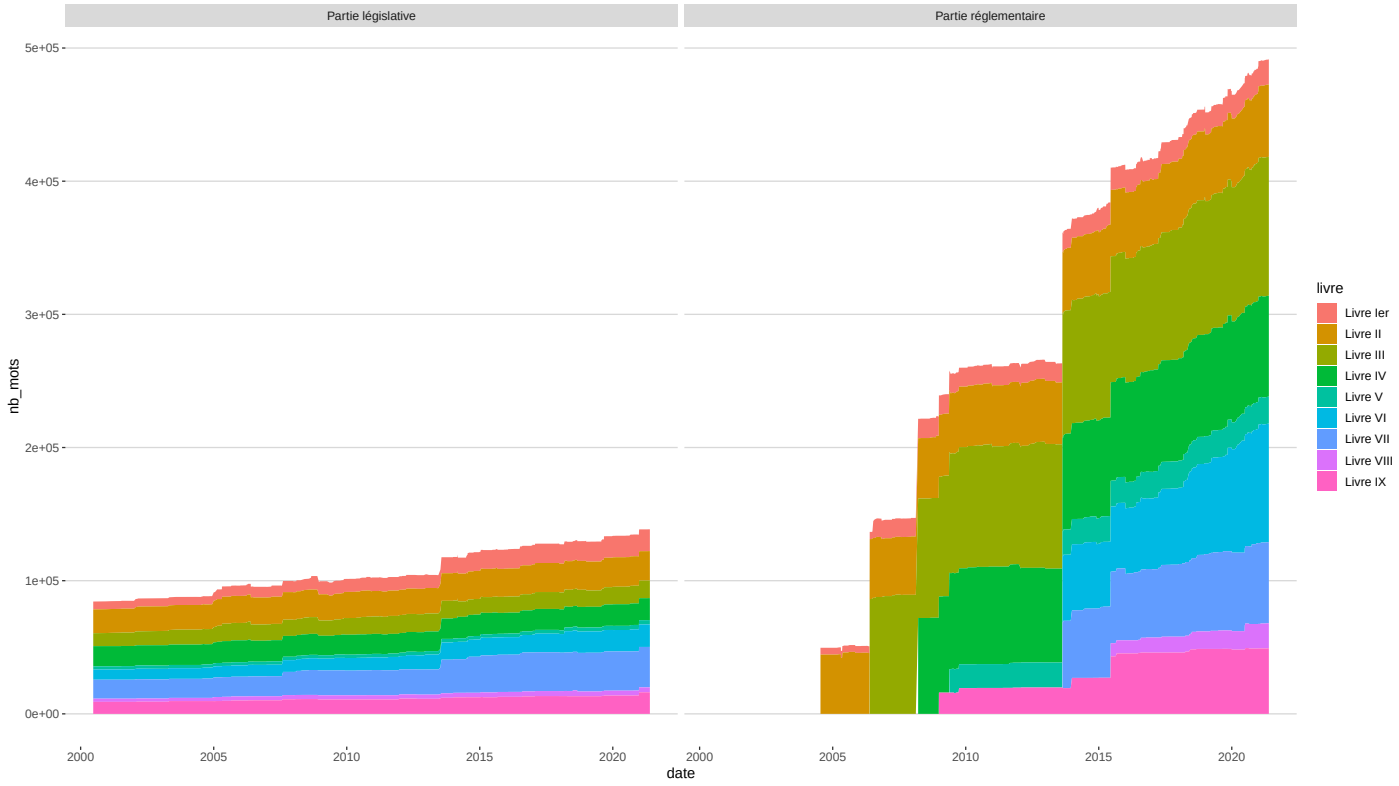
```
stats.det %>%
    mutate_at(
    c("partie","sous_partie","livre"),
    function(x) factor(x,unique(x))
    ) %>%
  ggplot(aes(x=date,y=nb_mots,fill=livre)) +
  geom_area(aes(group=livre)) +
  facet_grid(.~partie)  +
  theme_hc() +
  theme(legend.position = "right")
```



Deuxième problème : des titres changent au cours du temps.

Pour traiter ce problème, on va supprimer ces titres :

```
stats.det %>%
  mutate_at(
    c("sous_partie","livre"),
    function(x) gsub("(.*) :.*", "\\1",x)
    ) %>%
    mutate_at(
    c("partie","sous_partie","livre"),
    function(x) factor(x,unique(x))
    ) %>%
  ggplot(aes(x=date,y=nb_mots,fill=livre)) +
  geom_area(aes(group=livre)) +
  facet_grid(.~partie)  +
  theme_hc() +
  theme(legend.position = "right")
```



6.4.2 Code de la propriété intellectuelle