



Institut européen

des métiers de la **traduction** | IEMT

Université de Strasbourg

Web, corpus, traduction : exploitations

Sketch Engine, langage CQL

Enzo Doyen

2025 - M1

Langage CQL

Corpus **Q**uery **L**anguage : langage de requête pour interroger des corpus textuels.

Correspond à une chaîne de caractères exprimant un **motif linguistique** (un mot, ou une suite de mots) à partir des **valeurs de leurs propriétés** (catégorie grammaticale, lemme, forme graphique...).

Langage CQL

Utilisé par Sketch Engine, mais également par d'autres outils de recherche dans corpus (Frantext, la Base de Français Médiéval, entre autres).

Les requêtes CQL prennent également en charge les **expressions régulières**, un autre langage de requêtes (que nous verrons avec **AntConc**).



I. Tour d'horizon des expressions CQL

Principes de base

Les **guillemets** " " délimitent chaque item recherché.

"chat" : le mot **chat**

"chat noir" : la suite de mots **chat noir**

Principes de base

Les **crochets** [] peuvent être utilisés pour indiquer une propriété spécifique à rechercher, et sa valeur correspondante. Par exemple :

[word="présidents"] : le mot **présidents**

Distinction mot/lemme

Le **lemme** correspond à la forme canonique d'un mot variable.

Pour un nom, un adjectif ou un pronom, il s'agit de la forme au masculin singulier.

Par exemple, le lemme du mot « **princesses** » est « **princesse** ».

Pareillement, le lemme du pronom « **elles** » est « **il** ».

Distinction mot/lemme

Pour un verbe, le lemme est la forme à l'infinitif.

Par exemple, le lemme du mot « mangé » est « manger ».

Principes de base

Les **crochets** [] peuvent être utilisés pour indiquer une propriété spécifique à rechercher, et sa valeur correspondante. Par exemple :

[word="présidents"] : le mot **présidents**

[lemma="président"] : le lemme **président** (inclut les formes **président, présidente, présidents, présidentes**)

Principes de base : combinaison

On peut également combiner les items :

[lemma="le"] [lemma="président"] : recherche toutes les formes du mot **le** suivies de toutes les formes du mot **président**.

[lemma="président"] "de la République" : recherche toutes les formes du mot **président** suivies de la suite de mots **de la République**.

Principes de base : catégorie lexicale

Outre le mot et le lemme, on peut aussi spécifier la catégorie lexicale (POS, *part of speech*) :

[tag="N.*"] : les noms

[tag="V.*"] : les verbes

[tag="A.*"] : les adjectifs

etc.

Tableau des catégories lexicales principales

CQL	Catégorie lexicale
N.*	Noms
V.*	Verbes
A.*	Adjectifs
P.*	Pronoms
D.*	Déterminants
R.*	Adverbes
C.*	Conjonctions

**Aussi
disponibles
depuis
tableau
de bord
→
Corpus
Info**

COMMON TAGS

adjective	A.*
adverb	R.*
conjunction	C.*
determiner	D.*
interjection	I.*
noun	N.*
numeral	Z.*
preposition	S.*
pronoun	P.*
verb	V.*

[? All tags](#)

Principes de base : catégorie lexicale

Attention : les catégories lexicales et leurs abréviations peuvent changer selon le corpus sélectionné sur Sketch Engine.

Ils dépendent du **tagset** de la langue. Les exemples précédents et suivants prennent en considération le tagset *French FreeLing* : <https://www.sketchengine.eu/freeling-part-of-speech-tagset/>

Principes de base : opérateurs logiques

À l'intérieur des crochets, il est possible d'appliquer plusieurs conditions à l'aide des **opérateurs logiques** :

& : ET

| : OU

Principes de base : opérateurs logiques

[word="présidents" & tag="N.*"] : le mot **présidents** en tant que nom

[lemma="président" & tag="A.*"] : le lemme **président** en tant qu'adjectif

Principes de base : opérateurs logiques

[lemma="président|ministre|conseiller"] : le lemme **président**, **ministre** ou **conseiller**.

Principes de base : opérateurs logiques

❓ **Question :** que va donner la requête suivante ?

```
"un palais" [word="magnifique|beau|splendide"  
& tag="A.*"]
```

Principes de base : opérateurs logiques

❓ **Question :** que va donner la requête suivante ?

```
"un palais" [word="magnifique|beau|splendide"  
& tag="A.*"]
```

La suite de mots **un palais** suivi de l'adjectif **magnifique**, **beau** ou **splendide**.

Principes de base : laisser de l'espace

Parfois, on ne se rappelle plus exactement d'une expression, ou on souhaite voir quels mots pourraient s'intercaler entre deux autres.

Principes de base : laisser de l'espace

Parfois, on ne se rappelle plus exactement d'une expression, ou on souhaite voir quels mots pourraient s'intercaler entre deux autres.

C'est là qu'interviennent les **jokers**, ou crochets vides [].

Principes de base : laisser de l'espace

"un palais" [] "magnifique" : la suite de mots **un palais** suivie du mot **magnifique**, avec un mot entre les deux.

"ne casse pas" [] "pattes à un canard" : « ne casse pas X pattes à un canard »

Principes de base : laisser de l'espace

On peut laisser encore plus d'espace !

[lemma="il"] []{0,3} [lemma="y"] [lemma="avoir"]

0 : distance minimale

3 : distance maximale

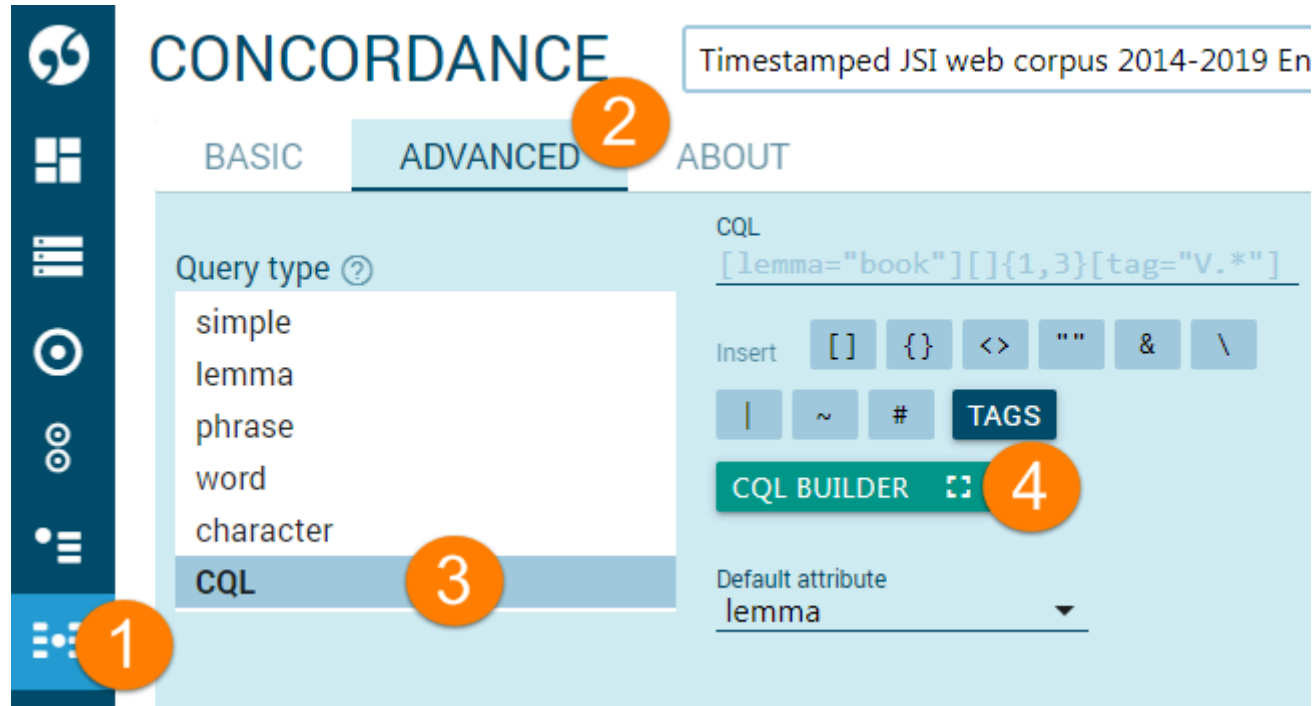
Principes de base : item facultatif

On peut indiquer qu'un item est facultatif avec le **point d'interrogation** ?.

[lemma="encodage"] []? [lemma="par"] []? [lemma="GPU"]

[word="première"] [word="ministre"] [word="française"]?

Constructeur de requêtes CQL de Sketch Engine¹



¹Source des images : <https://www.sketchengine.eu/news/cql-builder/>

Constructeur de requêtes CQL de Sketch Engine

The screenshot displays the Sketch Engine CQL builder interface. At the top, the Sketch Engine logo is on the left, and the title "CQL builder" is centered. To the right of the title are icons for saving, undo, redo, and help, followed by a close button (X). Below the title, the current CQL query is shown: `CQL: [tag="RB.?"] []? [tag="V.*"]`.

The main area shows the query construction process. It consists of three main components separated by plus signs:
1. A "normal token" block containing a list structure: `[Attribute? = Value "RB.?"]`. A dropdown menu is open over the "Value" field, showing a list of tags:

- N.* (noun)
- V.* (verb)
- J.* (adjective)
- RB.? (adverb)
- NN (noun)

 The "show all tags" link is also visible at the bottom of the dropdown.
2. An "any token" block containing the list structure: `[] ?`.
3. A "normal token" block containing the list structure: `[tag="V.*"]`.

Below the query construction area, there is a "result example" section. It shows a sample text snippet: "ith this job," Mujica said. </s><s> "Uruguay doe...". The text is annotated with red and blue tags. The red tags are "already been" and "also will need", and the blue tags are "affordably carry". The text is truncated on both sides.

On the right side of the interface, there is a red button labeled "USE THIS CQL" with a right-pointing arrow.

Pour aller plus loin

Tagset *French FreeLing* : <https://www.sketchengine.eu/freeling-part-of-speech-tagset/>

Documentation de Sketch Engine : <https://www.sketchengine.eu/documentation/cql-basics/>