

How to mark foreign words for accessibility in Drupal with the **CKEditor Language plugin**

Drupal offers a **native plugin**, learn
how to use it →

A common accessibility mistake related to foreign words

Authoring content almost always means reaching for loanwords. Writing in English and dropping in another language, or writing in any language and using anglicisms.

The failure is leaving those words **unmarked**, with no semantic hint about the language they belong to.

This maps to a specific rule: WCAG 3.1.2 "Language of Parts" (level AA), and RGAA 8.7. Since the European Accessibility Act came into force in June 2025, meeting WCAG AA **stopped being good practice and became a legal requirement.**

It breaks screen readers, and now the law

A blind user runs a screen reader set to English and reaches the word **paella**, unmarked.

The synthesizer applies English phonetics, something close to "*pay-ell-uh*". The reading breaks, and the word often becomes impossible to understand.

Beyond the experience, a failed criterion means **failing WCAG AA**, which is now the legal bar.

"pay-ell-uh"

The sound of an unmarked word.

Drupal already ships the fix

From Drupal 10 on, Core provides a native, simple answer to these criteria: the **CKEditor "Language"** selector plugin.

No *extra module*. No *complex dependency*. If you already have CKEditor, you already have the tool.

It marks each foreign word with the correct `lang` attribute. This plugin should not be missing from any site that takes accessibility seriously.

Why such an easy tool gets ignored

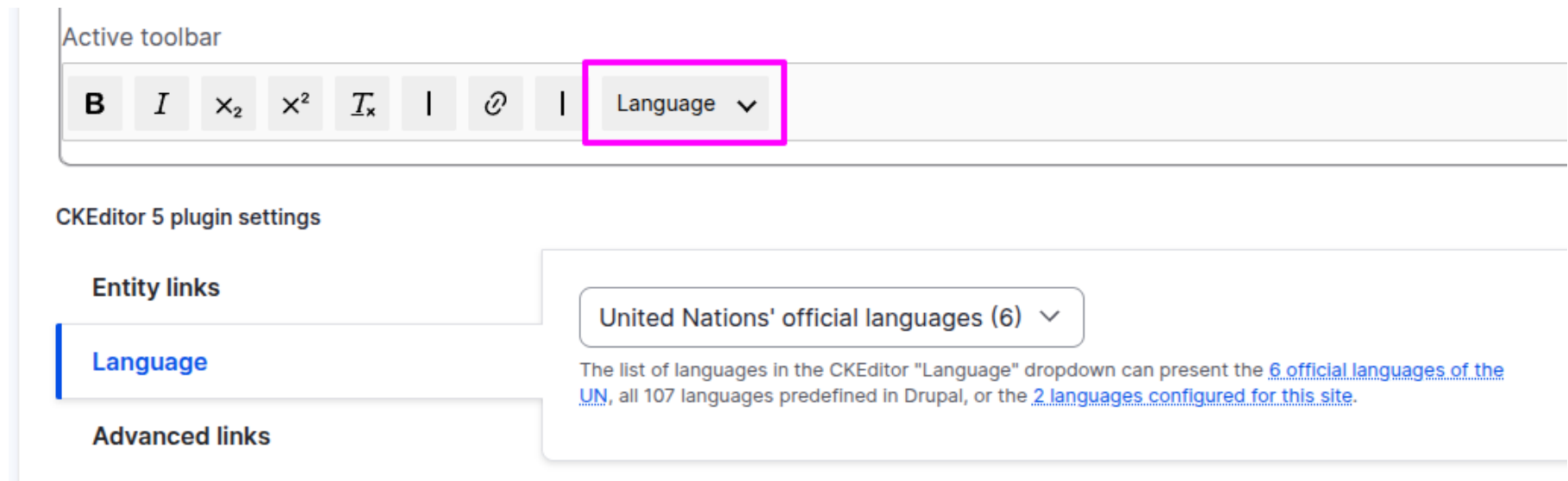
The plugin is trivial to use, yet it is skipped constantly. The reason is mostly **lack of awareness**.

Accessibility gained real weight only recently, so many technical profiles never learned about it. It does not help that the plugin is **not enabled by default** in Drupal's standard text formats. If nobody adds the button, nobody uses it.

How to set it up

THREE STEPS

1. Go to **Configuration > Content authoring > Text formats and editors**.
2. Drag the **Language** button into the active toolbar.
3. Scroll down, pick which languages appear in the dropdown, and save.



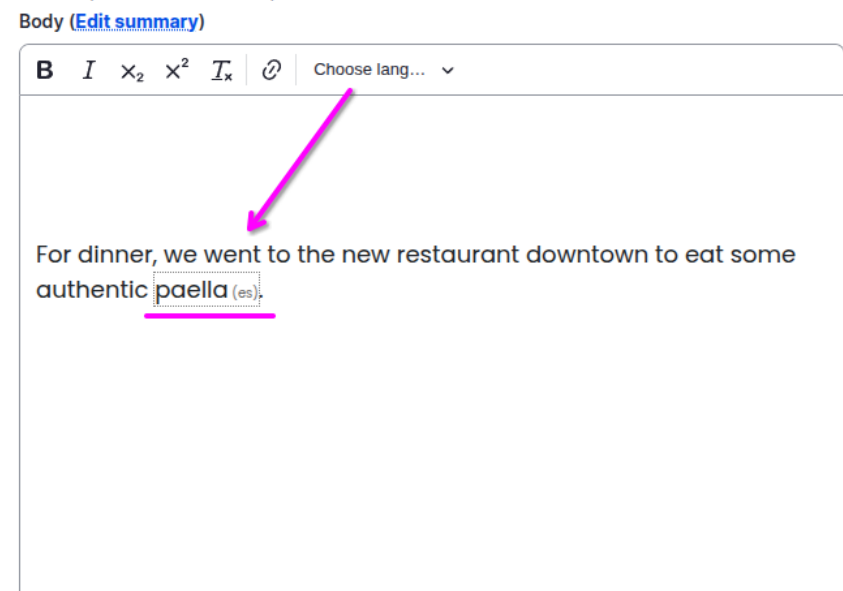
The selector is **independent** from your site's language system: show the 6 official UN languages, the 107 predefined in Drupal, or only those already configured.

How to use it

TWO ACTIONS

1. Write the text that needs a word in a language different from the current content.
2. Select the foreign word and pick the language it belongs to.

That is the whole workflow.
Each word is marked **manually** while writing.



The markup it produces

Select a word, pick its language, and the plugin wraps just that word in a `<span lang>`. It also adds `dir="ltr"` **automatically** (or `dir="rtl"` for Arabic or Hebrew).

ENGLISH TEXT, SPANISH WORD

```
... a great <span lang="es" dir="ltr">paella</span>.
```

SPANISH TEXT, ENGLISH WORD

```
... fue buen <span lang="en" dir="ltr">feedback</span>.
```

Compliance is one click away

Marking foreign words for accessibility sits one click away inside the editor itself. Drupal makes it native, both configuring and using it are very trivial, and the only real effort is the **habit** of marking words.

A tool this simple, native, and directly tied to a legal requirement leaves no good reason to skip it.



Making your Drupal site accessible?

At **Metadrop** we help teams meet WCAG and other accessibility criteria.

Follow us for more on Drupal, accessibility, and web development.



Save



Share



Comment

